

---

# **Kontaktregelung von Parallelrobotern auf der Basis von Aktionsprimitiven**

---

Thomas Reisinger



# **Kontaktregelung von Parallelrobotern auf der Basis von Aktionsprimitiven**

Von der Fakultät für Elektrotechnik, Informationstechnik und Physik  
der Technischen Universität Carolo-Wilhelmina  
zu Braunschweig

zur Erlangung der Würde

eines Doktor-Ingenieurs (Dr.-Ing.)

genehmigte Dissertation

von Dipl.-Ing. (FH) Thomas Reisinger  
aus Bad Soden-Salmünster

eingereicht am:	1. November 2007
mündliche Prüfung am:	7. März 2008
Referenten:	Prof. Dr.-Ing. W. Schumacher Prof. Dr.-Ing. Dr. h. c. E. Schnieder Prof. Dr.-Ing. W. Söte

2008



# Vorwort

DIESE Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Regelungstechnik an der Technischen Universität Braunschweig. Mein Dank gilt besonders Herrn Prof. Dr.-Ing. Walter Schumacher, der es mir ermöglichte im Rahmen des Sonderforschungsbereichs 562 – *Parallelroboter für Handhabung und Montage* in einer interdisziplinären Gruppe an einem aktuellen Forschungsthema zu arbeiten. Er unterstützte die Arbeit durch seine Anregungen und Bereitschaft zur Diskussion.

Für die Übernahme der Mitberichterstattung danke ich herzlich Prof. Dr.-Ing. Dr. h. c. Eckehard Schnieder vom Institut für Verkehrssicherheit und Automatisierungstechnik der TU Braunschweig. Danken möchte ich außerdem Prof. Dr. Werner Söte von der FH Osnabrück, für die Übernahme der Mitberichterstattung und dafür, dass er mir im Rahmen des *Assistentenprogramms für niedersächsische Fachhochschulen* die Möglichkeit zur Promotion eröffnet hat. Prof. em. Dr.-Ing. Dr. h. c. Werner Leonhard danke ich für die Übernahme des Vorsitzes der Prüfungskommission.

Ein herzliches Dankeschön geht an Dipl.-Ing. Michael Kolbus, der mich während meiner ganzen Zeit am Institut im gemeinsamen Büro begleitet hat, für die zahlreichen Diskussionen und aufmunternden Worte. Ihm, sowie den Dipl.-Ingenieuren Jochen Maaß und Christoph Budde vom Institut für Werkzeugmaschinen und Fertigungstechnik der TU Braunschweig, danke ich für die gemeinsame Implementierung der Steuerungsarchitektur und Inbetriebnahme des Parallelroboters HEXA. Ohne sie wäre das Gelingen des Projekts nicht möglich gewesen.

Andreas Rusniok danke ich für die prompte und sorgfältige Produktion der Maschinenelemente. Meinen ehemaligen Studenten Bernardo Bezerra, Luca Mura und Dipl.-Ing. Frank Wobbe bin ich für ihre Beiträge zur Simulations-Software sehr verbunden. Bei allen Mitarbeitern des Instituts bedanke ich mich für die vertrauensvolle Zusammenarbeit und offene Arbeitsatmosphäre.

Schließlich möchte ich mich bei meiner Familie bedanken, die mich immer förderte und meinen beruflichen Weg wohlwollend begleitete. Besonderer Dank geht an Susanne Schirmer. Sie hat mich auf meinem Weg in vielerlei Hinsicht und jederzeit unterstützt. In schwierigen Zeiten war sie ein sicherer Rückhalt und trug so wesentlich zum Gelingen dieser Arbeit bei.



# Kurzfassung

IN dieser Arbeit wird eine Kontaktregelung für Parallelroboter auf der Basis von Aktionsprimitiven vorgestellt. Durch den Einsatz von Aktionsprimitiven ist eine sensorbasierte und fehlertolerante Programmierung von Roboteraufgaben möglich. Die so vorgegebenen Regelziele in den einzelnen Achsen eines beliebig definierbaren Koordinatensystems werden durch Bewegungsmodule in eine Solltrajektorie umgesetzt und mit Hilfe des Task-Frame-Formalismus' in das Basiskoordinatensystem des Roboters transformiert. Eine unterlagerte Antriebsregelung sorgt dafür, dass der Endeffektor des Parallelroboters der Solltrajektorie möglichst genau folgt.

Um die unterlagerte Antriebsregelung den besonderen Eigenschaften des Parallelroboters anzupassen, ist sie als Kaskadenregelung in Gelenkkoordinaten mit innerem Geschwindigkeitsregelkreis und äußerem Lageregelkreis ausgeführt. Die Integration des inversen dynamischen Modells zur Berechnung der Vorsteuergrößen kapselt den nichtlinearen Parallelroboter und garantiert über den gesamten Arbeitsraum die Einhaltung der dynamischen Eigenschaften. Zusätzlich stellt sie eine kartesische Lageschnittstelle für den Task-Frame-Formalismus zur Verfügung. Die Stabilität der unterlagerten Antriebsregelung wird durch einen Beweis auf der Basis der Passivitätstheorie garantiert.

Der Begriff der Kontaktregelung umfasst in dieser Arbeit die Gesamtheit der Vorgänge während einer Montageaufgabe zwischen dem Endeffektor des Parallelroboters und der Umgebung. Die entwickelte Systemdynamik im Umgebungskontakt führt ein von der Umgebungsgeometrie abhängiges Kontaktkoordinatensystem ein und berücksichtigt so die auftretenden Kräfte und Momente im Task-Frame in einer realistischen Weise. Abpralleffekte bei der Aufnahme des Umgebungskontakts durch den Endeffektor werden mit den Gesetzmäßigkeiten der nicht-glaten Mechanik beschrieben.

Die Kontaktregelung ist als struktumschaltende Regelung in einem Bewegungsmodul realisiert. Sie umfasst einen Transitionsplaner und den Kraft-Momentenplaner. Der Transitionsplaner dämpft die Abpralleffekte während der Kontaktaufnahme und sorgt für einen stabilen Umgebungskontakt. Die Stabilität der verwendeten Algorithmen wird durch die Simulation mit *Moreaus Stoßprozess* validiert.

Die vorliegende Arbeit stellt damit erstmals ein umfassendes Kontaktregelungskonzept vor, das die Eigenschaften von Parallelrobotern und die Gesamtheit der im Umgebungskontakt auftretenden Effekte der nicht-glaten Mechanik für Robotersteuerungen auf der Basis von Aktionsprimitiven berücksichtigt.





# Abstract

IN this thesis, an interaction control concept for parallel kinematic machines based on robot programming with skill primitives is presented. The use of skill primitives allows a sensor-based and fault tolerant programming of robot tasks. Essential to this program paradigm is the task frame as a time varying orthogonal coordinate system. The origin and orientation of the frame is chosen at each instant such that, the task can be translated into a position or force control problem for a specific axis. Motion modules generate a reference trajectory that is transformed into the robot's base frame employing the so-called task frame formalism. A subordinate drive control ensure that the robot's end-effector follows the reference trajectory.

The subordinate drive control is designed as a cascade control in joint space with an inner velocity control loop and an outer position control loop because this structure suits the characteristics of parallel kinematic machines very well. Therefore, no calculation of the direct kinematics problem is necessary. The feed-forward signals are calculated by the inverse dynamics model. Hence, the subordinate drive control provides an interface to the task frame formalism in Cartesian space and encapsulates the robot and sensor hardware as a stable subsystem with uniform limits within the workspace. The stability of the subordinate drive control is verified using a passivity based approach.

A global approach describing all effects occurring while the end-effector establishes contact with a surface in the environment characterizes the interaction control concept. The constraint system dynamics introduces an environment dependent interaction frame. Thus, forces and torques in the task frame are modelled realistically during a robot task. Impacts of the robot's end-effector on the environment surface and the system behavior during the transition phase are described using non-smooth mechanics.

The interaction control concept employs a switched control law that is implemented in a motion module. It consists of a transition planner and a force-torque controller. The transition planner damps the end-effector impacts on the surface and establishes a stable environment contact. The stability of the algorithm is validated by simulations using *Moreau's sweeping process*.

The interaction control concept developed in this thesis, takes into consideration the special characteristics of parallel kinematic machines as well as non-smooth mechanics during the transition phase in robot control architectures implementing skill primitives for the first time.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Forschungsgegenstand und Motivation . . . . .	2
1.1.1	Kontaktregelung . . . . .	2
1.1.2	Parallelrobotik . . . . .	3
1.1.3	Roboterprogrammierung mit Aktionsprimitiven . . . . .	6
1.2	Stand der Forschung . . . . .	6
1.3	Stand der Technik . . . . .	8
1.4	Vorgehensweise . . . . .	9
<b>2</b>	<b>Robotersteuerung auf der Basis von Aktionsprimitiven</b>	<b>11</b>
2.1	Aktionsprimitive . . . . .	11
2.2	Steuerungsarchitektur . . . . .	15
2.2.1	Bewegungsmodule . . . . .	17
2.2.2	Task-Frame-Formalismus . . . . .	18
2.3	Antriebstechnik . . . . .	22
<b>3</b>	<b>Unterlagerte Antriebsregelung</b>	<b>23</b>
3.1	Kaskadenregelung . . . . .	23
3.1.1	Geschwindigkeitsregelung . . . . .	25
3.1.2	Lageregelung . . . . .	26
3.1.3	Auslegung der Kaskadenregelung . . . . .	26
3.1.4	Anpassung der Parameter des Geschwindigkeitsreglers . . . . .	28
3.2	Vorsteuerung . . . . .	29
3.3	Stabilität der Antriebsregelung . . . . .	32
3.3.1	Passivität . . . . .	32
3.3.2	Ruhelagen des Systems . . . . .	35
3.3.3	Ljapunow-Funktion . . . . .	36
3.3.4	Ableitung der Ljapunow-Funktion . . . . .	38
3.3.5	Stabilität der Geschwindigkeitsregelung . . . . .	40
3.3.6	Stabilität der Lageregelung . . . . .	42
3.4	Verhalten der Antriebsregelung . . . . .	45

<b>4</b>	<b>Systemdynamik im Umgebungskontakt</b>	<b>47</b>
4.1	Umgebungsgeometrie . . . . .	48
4.2	Kräfte und Momente . . . . .	50
4.2.1	Kräfte und Momente im Roboterkoordinatensystem . . . . .	52
4.2.2	Kräfte und Momente im Task-Frame . . . . .	52
4.3	Umgebungsmodell . . . . .	55
4.3.1	Kontaktmodell . . . . .	55
4.3.2	Transitionsmodell . . . . .	64
<b>5</b>	<b>Kontaktregelung</b>	<b>73</b>
5.1	Kontaktregelungskonzept . . . . .	73
5.1.1	Modellfolgeregelung . . . . .	75
5.1.2	Kraft-Momentenvorsteuerung . . . . .	77
5.1.3	Strukturumschaltende Regelung . . . . .	79
5.2	Kraft-Momentenregler . . . . .	81
5.2.1	Führungsgrößengenerator . . . . .	81
5.2.2	Kraft-Momentenregler mit statischem Umgebungsmodell . . . . .	84
5.2.3	Kraft-Momentenregler mit dynamischem Umgebungsmodell . . . . .	89
5.3	Transitionsplaner . . . . .	95
5.3.1	Kontaktaufnahme . . . . .	95
5.3.2	Dämpfung . . . . .	96
5.4	Stabilität und Sicherheit . . . . .	100
5.4.1	Aufgabenplanung . . . . .	100
5.4.2	Robotersteuerung auf der Basis von Aktionsprimitiven . . . . .	104
5.4.3	Stabilität der Kontaktregelung . . . . .	104
<b>6</b>	<b>Experimentelle Untersuchungen</b>	<b>107</b>
6.1	Versuchsaufbau . . . . .	107
6.2	Unterlagerte Antriebsregelung . . . . .	108
6.3	Kontaktregelung . . . . .	111
6.3.1	Funktionsprinzip . . . . .	111
6.3.2	Vergleich der Kontaktregler mit und ohne Dämpfungsphase . . . . .	113
6.3.3	Konturverfolgung . . . . .	115
6.3.4	Orthogonale und verkoppelte Regelziele . . . . .	116
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>119</b>
<b>A</b>	<b>Roboter</b>	<b>121</b>
A.1	Parallelroboter HEXA . . . . .	121
A.2	Serieller Roboter COMAU SMART . . . . .	122

---

<b>B</b>	<b>Passivitätstheorie</b>	<b>123</b>
B.1	Einführung und Definitionen . . . . .	123
B.2	Zusammenschaltung passiver Systeme . . . . .	124
B.3	Passivität von LTI-Systemen . . . . .	125
B.4	Passivität von Euler-Lagrange-Systemen . . . . .	126



# Symbolverzeichnis

Verzeichnis der im Text verwendeten mathematischen Symbole und Akronyme.

## Allgemeines

$a$	Skalar
$\mathbf{a}$	Vektor
$\mathbf{A}$	Matrix
$\mathbf{I}$	Einheitsmatrix
$f(x)$	Funktion von $x$
$\ \mathbf{a}\ $	Euklidische Norm des Vektors $\mathbf{a}$
$\ \mathbf{A}\ $	2-Norm der Matrix $\mathbf{A}$
$t$	Zeit

## Regelungstechnik und Parameter

$G(s)$	Übertragungsfunktion eines SISO-Systems
$\Sigma$	Zustandsraumdarstellung eines MIMO-Systems
$\mathbf{G}(s)$	Matrix mit Übertragungsfunktionen eines MIMO-Systems
$\{G(s)\}_n$	Matrix mit Übertragungsfunktionen eines MIMO-Systems. Das Mehrgrößensystem besteht dabei aus $n$ parallelen SISO-Systemen mit der gleichen Übertragungsfunktion $G(s)$
$G_K(s)$	Übertragungsfunktion eines Reglers
$\Sigma_K$	Zustandsraumdarstellung eines Mehrgrößenreglers
$\mathbf{K}$	Allgemeine Reglermatrizen
$V$	Verstärkungsfaktor
$T_x$	Zeitkonstante mit zusätzlicher Bezeichnung $x$
$V_i, T_{x,i}$	Verstärkungsfaktor (Zeitkonstante) des $i$ -ten Regelkreises
$V^{(k)}, T_x^{(k)}$	Bei Mehrgrößenregelungen: Verstärkungsfaktor (Zeitkonstante) des $k$ -ten Freiheitsgrads
$\mathbf{a}_{\text{fwd}}$	Vorsteuergrößen
$\mathbf{a}_{\text{ref}}$	Sollgrößen
$a_e$	Umgebungsgrößen
$\hat{a}$	Berechnete, geschätzte bzw. angenäherte Größen

$\bar{a}$	Mittelwert einer Größe
$\Delta$	Unsicherheitsfaktor
$A_R$	Regelfläche

## Mechanik

$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$	Antriebswinkel, Winkelgeschwindigkeit und Beschleunigung in Gelenkkoordinaten
$\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}$	Lage, Geschwindigkeit und Beschleunigung in kartesischen Koordinaten
$\mathbf{x} = [\mathbf{r}^T, \boldsymbol{\varphi}^T]^T$	Lage
$\mathbf{r} = [x, y, z]^T$	Ortsvektor
$\boldsymbol{\varphi} = [\varphi, \theta, \psi]^T$	Orientierung
$\mathbf{p}$	allgemeiner Vektor
$\mathbf{R}(\boldsymbol{\varphi}) = \mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\varphi)$	Rotationsmatrix. Darstellung in Roll-Pitch-Yaw-Konfiguration (RPY)
$\text{RPY}(\mathbf{R})$	Operator, der die RPY-Winkel der Rotationsmatrix $\mathbf{R}$ bestimmt
${}^B\mathbf{T}_A$	Homogene Transformation vom Koordinatensystem A in Koordinatensystem B
$\mathbf{D}$	Operator, der unter Anwendung von $\mathbf{D}^T \mathbf{T} \mathbf{D}$ die Rotationsmatrix der Transformation $\mathbf{T}$ liefert
$\mathbf{t}$	Operator, der unter Anwendung von $\mathbf{D}^T \mathbf{T} \mathbf{t}$ den Translationsvektor der Transformation $\mathbf{T}$ liefert
$\mathbf{s}$	Selektionsvektor $\mathbf{s} = [0, 0, 1]^T$
BF	Roboterkoordinatensystem, Bezugssystem (engl. base frame)
HF	Koordinatensystem des Endeffektors (engl. hand frame)
SF	Sensorkoordinatensystem (engl. sensor frame)
IF	Koordinatensystem des Kontaktpunkts (eng. interaction frame)
TF	Task-Frame, Koordinatensystem in dem die Regelungsaufgabe ausgeführt wird

## Robotik

$T_{\text{ref}}$	Trajektorie
$\mathbf{n}(\mathbf{x})$	Inverses kinematisches Problem. Berechnung der Antriebswinkel aus der Endeffektorlage
$\mathbf{h}(\mathbf{q})$	Direktes kinematisches Problem. Berechnung der Endeffektorlage aus den Antriebswinkeln



$\mathbf{J}_x$	Jacobimatrix in kartesischen Koordinaten
$\mathbf{J}_q$	Jacobimatrix in Gelenkkoordinaten
$\boldsymbol{\tau}$	Generalisierte Antriebsmomente in Gelenkkoordinaten
$\boldsymbol{\zeta} = [f_x, f_y, f_z]^T$	Kräfte in den translatorischen Freiheitsgraden eines Koordinatensystems
$\boldsymbol{\mu} = [\mu_\varphi, \mu_\theta, \mu_\psi]^T$	Drehmomente in den rotatorischen Freiheitsgraden eines Koordinatensystems
$\mathbf{f} = [\boldsymbol{\zeta}^T, \boldsymbol{\mu}^T]^T$	Generalisierte Kräfte in kartesischen Koordinaten
$\Gamma(\mathbf{r})$	Geometrie der Umgebung
$m$	Masse
$j$	Trägheitsmoment
$\mathbf{M}(\mathbf{x}), \mathbf{M}_x$	Massenträgheitsmatrix in $x$ -Koordinaten
$\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}), \mathbf{C}_x$	Matrix der Zentrifugal- und Corioliskräfte in $x$ -Koordinaten
$\mathbf{g}(\mathbf{x}), \mathbf{g}_x$	Gewichtskräfte in $x$ -Koordinaten

## Passivität und Regelungstheorie

$\Sigma$	System
$\mathbf{x}$	Zustandsvektor
$\mathbf{u}$	Eingangsvektor
$\mathbf{y}$	Ausgangsvektor
$\mathbf{A}$	Systemmatrix
$\mathbf{b}$	Eingangsvektor
$\mathbf{c}^T$	Ausgangsvektor
$d$	Durchgangsskalar eines Systems in Zustandsraumdarstellung
$\mathbf{r}$	Rückführungsvektor eines Zustandsreglers
$\mathbf{h}$	Korrekturvektor eines Beobachters
$w(\mathbf{u}, \mathbf{y})$	Versorgungsrate
$\mathcal{H}$	(Energie-)Speicherfunktion
$\mathcal{V}$	Potentielle Energie
$\mathcal{T}$	Kinetische Energie

## Moreaus Stoßprozess

$T$	Dauer der Transitionsphase
$t_0$	Beginn der Transitionsphase
$t_k, t_{k+1}, t_{k+2}, \dots$	Zeitpunkte des Umgebungskontakts in der Transitionsphase
$t_d$	Ende der Transitionsphase

$v$	Anzahl der Schritte zur Simulation
$h$	Zeitintervall
$\mathbf{u}$	Generalisierte Geschwindigkeit
$\mathbf{w}$	Gewichteter Durchschnitt der generalisierten Geschwindigkeit
$\mathbf{r}$	Reaktion
$\mathbf{p}$	Vektorfeld der Gewichts-, Coriolis- und Zentrifugalkräfte
$e$	Stoßzahl, Materialeigenschaft, die durch das Verhältnis von Annäherungs- zu Entfernungsgeschwindigkeit während eines Stoßes mit einem Probekörper bestimmt wird
$k = \frac{e}{1+e}$	Alternative Formulierung der Stoßzahl
$L$	Freier Arbeitsraum
$V$	Tangentialkegel
$N_V$	Normalenkegel

## Abkürzungen und Akronyme

DAE	Differential-algebraische Gleichung (engl. differential algebraic equation)
I	Integrator
ITAE	Zeitgewichtete Betragsregelfläche (engl. integral of time multiplied absolute error)
LTI	Lineares, zeitinvariantes System (engl. linear time-invariant system)
MIMO	Mehrgrößensystem (engl. Multiple input, multiple output)
P	Verstärker, Proportionalelement
PI	Proportional-Integral-Element
PT <sub>1</sub>	Verzögerungsglied erster Ordnung
PT <sub>2</sub>	Verzögerungsglied zweiter Ordnung
PT <sub>1</sub> D	Proportionalelement mit Verzögerung erster Ordnung und Vorhalt
SISO	Eingößensystem (engl. Single input, single output)

# Kapitel 1

## Einleitung

DER in den letzten Jahrzehnten zu beobachtende Trend hin zur Automatisierung der Produktion ist ungebrochen. Roboter, als universell einsetzbare Werkzeuge, sind dabei unverzichtbarer Bestandteil dieser Entwicklung. Nach Grundler [2004] ist und bleibt die Robotik damit ein wichtiger volkswirtschaftlicher Wettbewerbsfaktor. Neben dem Ausbau der bestehenden Märkte kommt der Erschließung neuer Anwendungsbereiche eine zentrale Bedeutung zu [Karlsson, 2005].

Nach einer Studie des VDMA<sup>1</sup> [vgl. VDMA, 2002] entfallen 70% der Produktkosten auf Handhabungs- und Montageprozesse. Der Einsatz von geeigneten Robotern würde zu einer Verkürzung der Taktzeiten bei gleichzeitiger Erhöhung der Prozessqualität führen und damit zu einer Produktivitätssteigerung in der Produktion. Besonderes Potenzial liegt in der Anwendung von Parallelrobotern, die in einigen Teilbereichen deutliche strukturbedingte Vorteile gegenüber den heute in der Industrie hauptsächlich eingesetzten seriellen Robotern besitzen: Durch die gestellfeste Anordnung der Antriebe sind die bewegten Teile leichter und besitzen damit im Allgemeinen eine höhere Dynamik als serielle Roboterstrukturen. Letztere sind im Aufbau dem menschlichen Arm nachempfunden und führen die Antriebe in der kinematischen Kette mit sich, so dass ein Antrieb zusätzlich zu dem Gewicht der Glieder alle nachfolgenden Antriebe tragen muss. Der Aufbau aus mehreren kinematischen Ketten verleiht Parallelrobotern zudem eine höhere Steifigkeit und Genauigkeit. Während zur Zeit kommerziell verfügbare Parallelroboter fast ausschließlich für Handhabungsaufgaben eingesetzt werden, beschäftigt sich die vorliegende Arbeit mit der Entwicklung von Regelungskonzepten zur Hochgeschwindigkeitsmontage mit Parallelrobotern.

Das Einsatzgebiet der Hochgeschwindigkeitsmontage befindet sich in den Schnittpunkten der Handhabung, Leiterplattenbestückung und Präzisionsmontage. Es hat mit der Handhabung gemein, dass die Taktzeiten möglichst kurz sein sollen, grenzt sich gegenüber dieser jedoch ab, da Werkstücke nicht nur transportiert, sondern auch montiert werden. Gegenüber der Leiterplattenbestückung werden größere Teile bewegt und in komplexeren Bewegungsabläufen montiert. Im Vergleich zur Präzisionsmontage sind die Arbeitszyklen kürzer, allerdings sind nicht so enge Toleranzen zu beachten.

---

<sup>1</sup>Verband deutscher Maschinen- und Anlagenbau e.V.

## 1.1 Forschungsgegenstand und Motivation

Die Anwendung von Parallelrobotern zur Hochgeschwindigkeitsmontage in der industriellen Produktion erfordert das Vorhandensein einer zuverlässigen Prozesskette von der Roboterprogrammierung über die Steuerungsfunktionen bis zur Kontakt- und Lageregelung. In diesem Kontext stellen sowohl die übergeordnete Roboterprogrammierung als auch die Eigenschaften der Parallelroboter Anforderungen an die Kontaktregelung. Zusammen mit einer umfassenden Betrachtung der Kontaktregelung legen sie den Forschungsgegenstand dieser Arbeit fest.

### 1.1.1 Kontaktregelung

Klassische Ansätze zur Kraftregelung gehen davon aus, dass sich der Endeffektor des Roboters zu Beginn der Regelungsaufgabe im Kontakt mit der Umgebung befindet. Demgegenüber stellt das Konzept der Kontaktregelung eine Erweiterung dar, das alle während der Interaktion des Endeffektors mit der Umgebung auftretenden Effekte in ihrer Gesamtheit betrachtet. Eine Kontaktregelung kann dabei nach Brogliato [1999] in die drei in Abbildung 1.1 dargestellten Phasen eingeteilt werden:

- Phase 1: Sie beginnt mit der Annäherung des Endeffektors an die Kontaktfläche. In dieser Phase der *Freiraumbewegung* findet jedoch kein Umgebungskontakt statt sondern die Bewegung stoppt kurz vorher. Etwaige Unsicherheiten in der Beschreibung der Geometrie können so berücksichtigt werden.
- Phase 2: In der *Transitionsphase* berührt der Endeffektor die Umgebung. Dabei prallt er aufgrund einer zu großen kinetischen Energie möglicherweise ab und verliert den Kontakt wieder. Dieser Vorgang kann sich mehrmals wiederholen. Die in dieser Phase im Eingriff befindlichen Regler müssen die Abpralleffekte dämpfen und einen sicheren Umgebungskontakt gewährleisten.
- Phase 3: Erst wenn der Umgebungskontakt dauerhaft hergestellt ist, gelangt der Kraft-Momentenregler in den Eingriff und stellt die gewünschte Endeffektorkraft bzw. das gewünschte Moment ein. Diese so genannte *Kontaktphase* endet, wenn der Endeffektor die Umgebung wieder verlässt.

Nur durch diese Betrachtungsweise und eine entsprechende Auslegung der Regler in den einzelnen Phasen ist die asymptotische Stabilität der Kontaktregelung zu gewährleisten [Brogliato, 1999].

Im Rahmen dieser Arbeit wird immer explizit von einer Kraft- und Momentenregelung gesprochen, da sich die translatorischen und rotatorischen Freiheitsgrade sowohl in der Beschreibung der Systemdynamik im Umgebungskontakt als auch in der Reglerauslegung unterscheiden.

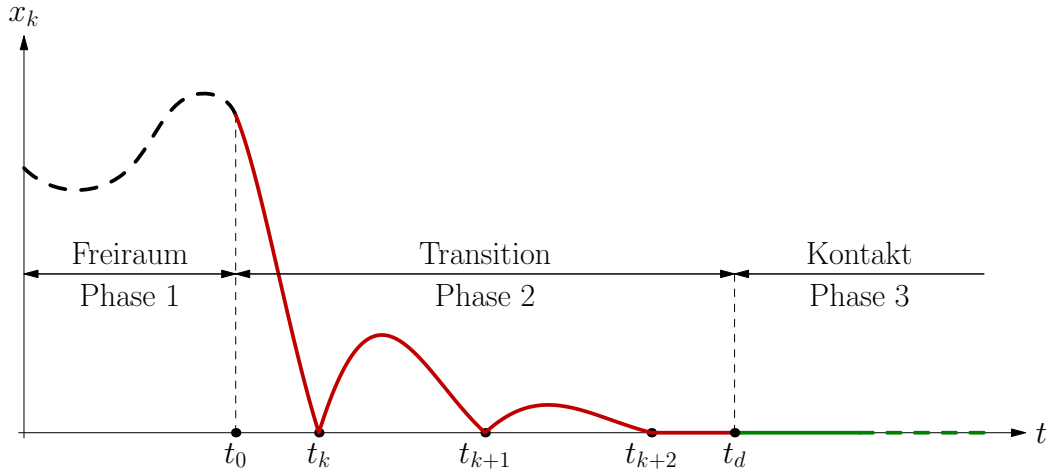


Abb. 1.1: Phasen der Kontaktregelung

### 1.1.2 Parallelrobotik

Parallelroboter sind, genau wie serielle Roboter, programmierbare Automatisierungswerkzeuge, die Handhabungs- und Montageaufgaben ausführen können. Durch die Möglichkeit zur Programmierung sind sie flexibel einsetzbar und an verschiedene Aufgaben anzupassen. Serielle Roboter, deren Strukturbild in Abbildung 1.2(a) dargestellt ist, bestehen aus einer offenen kinematischen Kette, die vom Gestell bis zum Endeffektor führt. Die Antriebe befinden sich in den Gelenken und werden mitgeführt. Im Gegensatz dazu sind die Antriebe des in Abbildung 1.2(b) dargestellten Parallelroboters gestellfest montiert. Mehrere kinematische Ketten führen vom Gestell über passive Gelenke parallel zum Endeffektor. Da sich die Last am Endeffektor auf mehrere Führungsketten verteilt und keine Antriebe mitgeführt werden, ist es möglich, die Struktur von Parallelrobotern leichter zu bauen. Daraus resultiert eine höhere Dynamik bei gleicher Steifigkeit der Struktur. Parallelroboter können trotz ihres geringen Gewichtes große Lasten tragen. Das erreichbare Verhältnis von Nutzlast zu Eigengewicht kann in manchen Fällen größer als eins sein. Durch die parallele Anordnung der kinematischen Ketten erhöht sich die erreichbare Genauigkeit der Endeffektorlage. Messfehler und Fertigungstoleranzen summieren sich nicht über eine Kette, sondern können ausgeglichen werden [vgl. Merlet, 2000; Tsai, 1999]. Der in Abbildung 1.3(a) gezeigte Parallelroboter HEXA<sup>2</sup> mit sechs Freiheitsgraden realisiert die oben beschriebenen Konzepte.

Den oben beschriebenen Vorteilen stehen folgende Nachteile gegenüber: Die hohe Anzahl der Führungsketten, die vom Gestell zum Endeffektor führen, schränkt die Beweglichkeit besonders der rotatorischen Freiheitsgrade ein, da sich die Gefahr der Eigenkollision erhöht. Singularitäten befinden sich auch innerhalb des Arbeitsraums, während diese bei seriellen Strukturen nur am Rand auftreten. Diesen Eigenschaften muss durch besondere maschinen-

<sup>2</sup>Der Parallelroboter HEXA dient als Versuchsträger für die Erprobung und Validierung der Kontaktregelung. Eine detaillierte Beschreibung findet sich im Anhang A.1.

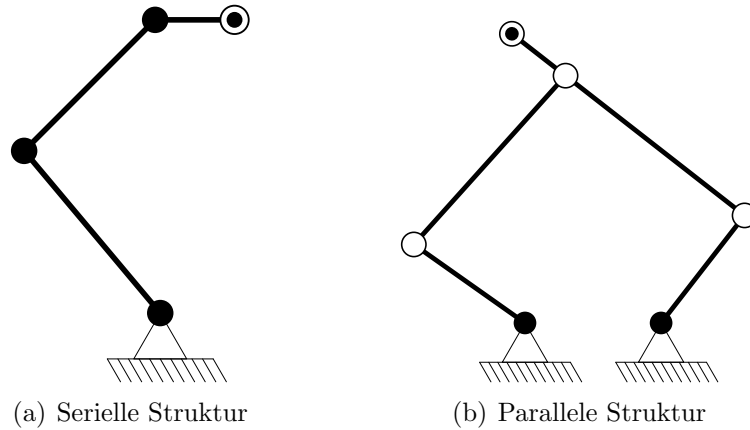


Abb. 1.2: Prinzipskizze der verschiedenen Robotertypen



(a) HEXA



(b) TRIGLIDE

Abb. 1.3: Verschiedene Realisierungen von Parallelrobotern

nahe Steuerungsfunktionalitäten zur Arbeitsraumüberwachung Rechnung getragen werden. Durch die Anwendungen hybrider Konzepte, die sowohl geschlossene als auch offene kinematischen Ketten enthalten, sollen diese Nachteile überwunden werden. Der in Abbildung 1.3(b) dargestellte Roboter TRIGLIDE besitzt drei Linearantriebe und kinematische Ketten, die die Bewegung in den translatorischen Freiheitsgraden realisieren. Ein weiterer in der Arbeitsplattform montierter Antrieb sorgt für einen zusätzlichen rotatorischen Freiheitsgrad, ohne die Bewegung der Struktur weiter einzuschränken.

Aus regelungstechnischer Sicht betrachtet, weisen Parallelroboter einige Besonderheiten auf. Zwar handelt es sich bei diesen Systemen ebenfalls um nichtlineare Mehrgrößensysteme. Die Terme der Zentrifugal- und Corioliskräfte sind jedoch ausgeprägter vorhanden, so dass die Beherrschung hochdynamischer Bewegungen nur durch modellbasierte Regelungsmethoden möglich ist. Die Besonderheiten sollen im Folgenden kurz dargestellt werden. Wie in Abbildung 1.4 aufgezeigt, ist die Besetzung der Massenträgheitsmatrix wesentlich inhomogener. In beiden Diagrammen sind die Elemente der Massenträgheitsmatrix  $\mathbf{M}(\mathbf{x})$  für eine typische Pose im Zentrum des Arbeitsraums aufgetragen. Die  $x$ - und  $y$ -Achse reprä-

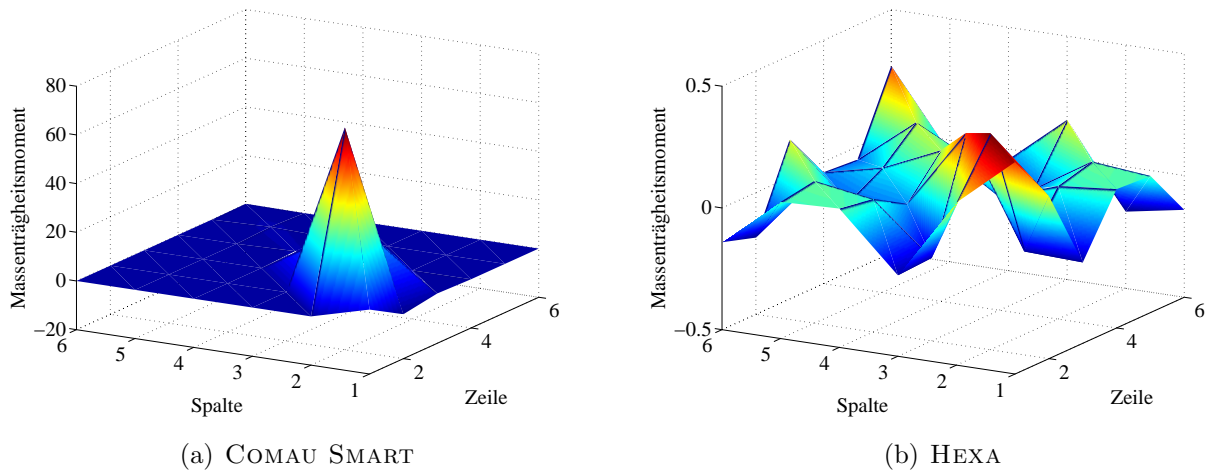


Abb. 1.4: Grafische Darstellung der Massenträgheitsmatrizen

sentieren die Zeilen bzw. Spalten des Tensors, während die farbliche Skalierung und die  $z$ -Achse den Wert des jeweiligen Elements anzeigt. Zwischen den Elementen wurden die Werte interpoliert, um eine geschlossene Fläche zu erhalten. Die Massenträgheitsmatrix des seriellen Roboters COMAU SMART<sup>3</sup> in Abbildung 1.4(a) wird von den Werten auf der Hauptdiagonalen dominiert. Gegenüber den Massenträgheitsmomenten der translatorischen Freiheitsgrade sind die Momente der rotatorischen Freiheitsgrade und die Kreuzkopplungen auf den Nebendiagonalen zu vernachlässigen. Eine gänzlich andere Konfiguration zeigt die Massenträgheitsmatrix des Parallelroboters HEXA in Abbildung 1.4(b). Der Maximalwert ist wesentlich kleiner als im Fall des seriellen Roboters. Jedoch befinden sich alle Werte in der gleichen Größenordnung und sind teilweise negativ. Sowohl die Elemente der rotatorischen Freiheitsgrade als auch die Kreuzkoppelterme sind signifikant besetzt.

Neben diesen Unterschieden in der Massenträgheitsmatrix weichen beide Robotertypen auch in den Dämpfungseigenschaften signifikant voneinander ab. Die durch die Coriolis-Matrix in das System eingebrachte Dämpfung ist bei seriellen Strukturen wesentlich größer. Untersuchungen im Rahmen dieser Arbeit haben für den Roboter COMAU SMART zehnmal größere Werte ergeben als für die Parallelkinematik HEXA. Diesem Umstand wird durch die Einführung eines Stabilitätsbeweises der Antriebsregelung auf Basis der Passivitätstheorie Rechnung getragen.

Parallelroboter besitzen im Allgemeinen ein analytisch lösbares inverses kinematisches Problem. Demgegenüber ist die Lösung des direkten kinematischen Problems nicht eindeutig und kann für komplexe Parallelstrukturen wie den HEXA ohne zusätzliche Gelenksensoren nur numerisch iterativ gelöst werden [Hesselbach u. a., 2004]. Dies schränkt die Anwendung von modellbasierten Regelungskonzepten, die auf einer Berechnung des direkten

<sup>3</sup>Der serielle Roboter COMAU SMART dient im Verlauf der Arbeit als Vergleichsobjekt. Eine detaillierte Beschreibung inklusive dem dynamischen Modell findet sich im Anhang A.2.

kinematischen Problems im Regelungsstakt beruhen, ein [Reisinger, 2005]. Statt dessen wird in dieser Arbeit ein Regelungskonzept vorgeschlagen, das auf Integration des Modells in der Berechnung der Vorsteuergrößen beruht.

### 1.1.3 Roboterprogrammierung mit Aktionsprimitiven

Die Programmierung von Roboteraufgaben mit Bewegungen im Umgebungskontakt stößt bei der Anwendung prozeduraler Sprachen oder Teach-in-Verfahren auf Schwierigkeiten. In der Transitionsphase der Kontaktregelung ist die Lage des Werkstücks und der Umgebung eine kritische Größe. Aufgrund von Fertigungstoleranzen und Ungenauigkeiten beim Greifvorgang ist die Kenntnis der aktuellen Geometrie nur unzureichend gegeben. Während einer kraft- oder momentengeführten Roboterbewegung ist die Lage des Endeffektors zum Zeitpunkt der Programmerstellung nicht bekannt. Es ist nicht möglich während des Produktionsprozesses auf diese sich verändernde Größen zu reagieren.

Dagegen berücksichtigt das von Hasegawa u. a. [1992] eingeführte und von Finkemeyer u. a. [2005a] aufgenommene Konzept der Roboterprogrammierung auf der Basis von Aktionsprimitiven diese Anforderungen. Dabei werden komplexe Roboteraufgaben in atomare Teilaufgaben, die so genannten Aktionsprimitive, zerlegt. Für jedes Aktionsprimitiv kann das Koordinatensystem, in dem die Aufgabe ausgeführt werden soll, vorgegeben werden. In jeder Achse dieses Koordinatensystems kann wiederum eine Regelungsaufgabe festgelegt werden, so dass letztendlich eine hybride Regelung<sup>4</sup> in beweglichen Koordinaten realisiert wird. Die Vernetzung mehrerer Aktionsprimitive in einem gerichteten Graphen erlaubt die Definition komplexer Roboteraufgaben. Zur Umschaltung zwischen den Aktionsprimitiven können sowohl Zustandsgrößen des Roboters als auch Sensormesswerte verwendet werden, so dass Toleranzen und die aktuelle Roboterpose während der Ausführung berücksichtigt werden. In Thomas u. a. [2004] ist eine Montageplanung beschrieben, die aus den CAD-Daten automatisiert Aktionsprimitivnetze generiert.

Aufgrund dieser Eigenschaften wird die Roboterprogrammierung mit Aktionsprimitiven bei der Entwicklung einer Kontaktregelung für Parallelroboter in dieser Arbeit vorausgesetzt und die Konzepte darauf ausgerichtet.

## 1.2 Stand der Forschung

Die hybride Lage- und Kraftregelung beruht auf der Ein-Ausgangs-Linearisierung der nicht-linearen Regelstrecke des Roboters und der Messung der Kontaktkräfte, so dass von einem

---

<sup>4</sup>Zur Abgrenzung des beschriebenen Verfahrens gegenüber anderen Regelungskonzepten mit gleichem oder ähnlichem Namen sei bemerkt, dass mit *hybrid* in diesem Fall die gleichzeitige Anwendung unterschiedlicher Regler für verschiedene Koordinatenachsen und gemeint ist. Hier ist nicht die Rede von Regelungskonzepten, die kontinuierliche und diskrete Systeme vereinen und die Bezeichnung *hybrid* auf die Integration dieser beiden Eigenschaften beziehen.



direkten Kraftregelungsverfahren gesprochen wird. Die in der Literatur beschriebenen Verfahren basieren auf der Arbeit von Raibert und Craig [1981] sowie ihren Erweiterungen [vgl. Canudas de Wit u. a., 1996; Sciavicco und Siciliano, 2000] und sind Grundlage zahlreicher Weiterentwicklungen. Die Arbeiten von Oya u. a. [2003] sowie Wang u. a. [2004] erweitern den Ansatz für eine unsichere Umgebungsgeometrie und nicht-holonome Randbedingungen. Diesen Verfahren ist gemein, dass sie die Lösung des direkten kinematischen Problems erfordern, welches für Parallelroboter im Allgemeinen nicht analytisch lösbar vorliegt.

Hogan [1985] begründete die Entwicklung von indirekten Kraftregelungsverfahren. Diese so genannten Impedanz- bzw. Admittanzregelungen regeln die vom Endeffektor ausgeübte Kraft durch die Vorgabe eines Positionssollwertes unter Berücksichtigung der Umgebungssteifigkeit, so dass sich die gewünschte Kraft einstellt. Der Ansatz ist robust im Bezug auf ungenaue Umgebungsbeschreibungen, die Kontaktkraft kann bei unbekannter Umgebungssteifigkeit allerdings nur mit großen Abweichungen eingestellt werden. In Seul Jung u. a. [2004] wird dieser Nachteil durch ein Adaptionsgesetz eliminiert. Die erwähnten Regelungskonzepte erfordern ebenfalls die Lösung des direkten kinematischen Problems und sind daher aus den genannten Gründen für Parallelroboter nicht geeignet.

Im Unterschied zu den oben beschriebenen Regelungskonzepten integriert die Methode der berechneten Momente [vgl. Sciavicco und Siciliano, 2000] bzw. die flachheitsbasierte Regelung nach Rothfuß u. a. [1997] das dynamische Modell in den Vorsteuerungszeit. Wie Whitcomb u. a. [1993] zeigte, führt diese Vorgehensweise bei unsicheren Modellparametern zu geringeren Regelfehlern und wird in der in dieser Arbeit vorgestellten unterlagerten Antriebsregelung für Parallelroboter angewandt.

Die Integration der Verzögerung durch die Antriebe wird durch die Kaskadenregelung [vgl. Leonhard, 1990] ermöglicht. Die Implementierung in Gelenkkoordinaten benötigt ausschließlich die Lösung des inversen kinematischen Problems.

Ein Regelungskonzept, das explizit die im System vorhandene Energie beeinflusst, beruht auf der Passivitätstheorie [vgl. Khalil, 2002; Kugi, 2005; Kugi und Schlacher, 2002; van der Schaft, 2000]. Die Anwendung auf Euler-Lagrange-Systeme und allgemeine nichtlineare Regelstrecken der so genannten passivitätsbasierten Regelung wird in Ortega u. a. [1998] sowie Sepulchre u. a. [1997] diskutiert. Darüber hinaus sind die Ansätze der Roboterregelung nach Arimoto darauf zurückzuführen [vgl. Arimoto, 2000; Cheah u. a., 2003]. Nach Loría [2004] gelang damit erstmals der Stabilitätsbeweis einer Kaskaden-Antriebsregelung eines nichtlinearen Mehrgrößensystems [vgl. Panteley und Ortega, 1997]. Für den DLR-Leichtbauroboter stellen Ott u. a. [2005] einen Stabilitätsbeweis einer Impedanzregelung unter Berücksichtigung der Gelenkelastizitäten vor. Die Passivitätstheorie stellt ein universelles Werkzeug dar, das den Stabilitätsbeweis für hochgradig nichtlineare Mehrgrößensysteme, wie sie Parallelroboter darstellen, ermöglicht.

Schon in der wegweisenden Arbeit von McClamroch und Wang [1988] wird die Notwendigkeit der korrekten Beschreibung der Umgebungsgeometrie für die Stabilität der Kraftregelung beschrieben. Vukobratovic und Tuneski [1994] sowie Yoshikawa [2000] erweiterten

diese Darstellung im Rahmen einer Stabilitätsuntersuchung der hybriden Lage- und Kraftregelung. Eine ungenaue Beschreibung der Umgebungsgeometrie führt danach zu Verkoppelungen in den unterschiedlichen Koordinatenachsen, was wiederum instabile Regelkreise zur Folge haben kann. Die Entwicklung einer Systemdynamik im Umgebungskontakt für eine Steuerungsarchitektur auf der Basis von Aktionsprimitiven im Rahmen dieser Arbeit berücksichtigt diese Tatsache.

Die Bedeutung der Transitionsphase für die Stabilität und Qualität einer Kontaktregelung wird in Brogliato [1999] beschrieben. Die Arbeiten von Pagilla und Biao [2001], Mills und Lokhorst [1993] sowie Tornambe [1999] entwickeln Regelungskonzepte für Roboter und allgemeine Euler-Lagrange-Systeme, die die Gesetzmäßigkeiten der nicht-glaten Mechanik berücksichtigen. Dagegen schaltet der von Chiaverini und Sciavicco [1993] vorgestellte parallele Ansatz zwar selbständig zwischen den verschiedenen Reglern zur Annäherung des Endeffektors an die Oberfläche und zur Kraftregelung um, vernachlässigt aber die Dämpfung der Abpralleffekte während der Transitionsphase.

Kunze und Monteiro Marques [2000] sowie Acary und Brogliato [2005] diskutieren die Anwendung von *Moreaus Stoßprozess* zur Simulation multilateraler Umgebungskontakte in der Transitionsphase. Ein Alleinstellungsmerkmal der vorliegenden Arbeit ist die Entwicklung einer Kontaktregelung für eine Steuerungsarchitektur auf der Basis von Aktionsprimitiven, die das Systemverhalten während der Transitionsphase in den Stabilitätsbeweis integriert. Eine weitere Abgrenzung gegenüber Finkemeyer [2004] und Kröger u. a. [2004] besteht in der Anwendung des erweiterten 'Task-Frame-Formalismus' sowie der Vorsteuerung der erwarteten Kontaktkräfte [vgl. Maaß u. a., 2006].

## 1.3 Stand der Technik

Der Einsatz von Parallelrobotern in der Industrie beschränkt sich auf wenige kommerziell verfügbare Geräte, die hauptsächlich für Handhabungsaufgaben eingesetzt werden. Nach dem Auslaufen diverser Patente in den nächsten Jahren wird jedoch mit einer Belebung des Marktes und einer weiteren Verbreitung dieses Robotertyps gerechnet.

Moderne Robotersteuerungen bestehen aus einem Leistungsteil, der die Motorumrichter enthält, und der Antriebsregelung. Zusätzliche Sicherheits- und Überwachungsfunktionen gewährleisten einen gefahrlosen Betrieb und verhindern, dass der Roboter zerstört oder Menschen verletzt werden. Die Antriebsregelung ist in Kaskadenstruktur aufgebaut, so dass, beginnend mit dem Stromregler, diesem ein Geschwindigkeitsregler und diesem wiederum ein Lageregler übergeordnet ist. Führungsgrößengeneratoren zur Bahnplanung sorgen für die synchrone Bewegung der Roboterachsen und die Berücksichtigung der dynamischen Eigenschaften der Regelstrecke. Dem Anwender steht eine herstellerspezifische Schnittstelle zur Verfügung. Mit Hilfe einer prozeduralen Programmiersprache oder durch ein Teach-in-Verfahren werden bekannte Bahnpunkte vorgegeben und so die Programmierung einer Bewe-

gungsaufgabe ermöglicht. Daneben existieren Betriebsmodi, die dem Anwender den direkten Zugriff auf die Lage- oder Geschwindigkeitsschnittstelle der Robotersteuerung ermöglichen. Diese Schnittstellen können durch Analogsignale oder Feldbussysteme angesprochen werden und ermöglichen so die Implementierung anwendungsspezifischer Regelungen wie z. B. einer Kraftregelung.

Obwohl schon in den achtziger Jahren von Raibert und Craig [1981] sowie von Slotine und Asada [1986] die Grundlagen der direkten bzw. indirekten Kraftregelung erforscht wurden, bleibt die industrielle Anwendung auf einzelne individuelle Anwendungen beschränkt. Die vielfältigen Konfigurationsmöglichkeiten der Regelung, die unsicheren Parameter der Umgebung und die Toleranzen der Werkstücke erfordern einen langwierigen Inbetriebnahmeprozess und wirken sich nachteilig auf die Langzeitstabilität der Kraftregelung aus. Erst in den letzten Jahren bieten verschiedene Hersteller Pakete mit integrierten Kraftregelungsalgorithmen und Schnittstellen zum Anschluss der entsprechenden Sensoren für ihre Robotersteuerungen an: Die Robotersteuerungssoftware *RobotWare Assembly FC* der Firma ABB ermöglicht die Durchführung von Montageaufgaben mit seriellen Robotern. Außerdem bietet die Firma Stäubli ein Softwarepaket namens *Compliance* an, dass nach Herstellerangaben zur Regelung externer Kräfte am Endeffektor des Roboters geeignet ist. In der Veröffentlichung [FANUC Robotics, 2005] der Firma FANUC wird eine Roboter montage vorgestellt und dabei auf eine nicht näher benannte *spezielle* Kraftregelungssoftware verwiesen.

Ein Grund für den geringen Einsatz der kraft- bzw. momentengeführten Montage in der industriellen Fertigung ist auch in der Roboterprogrammierung zu sehen. Die bisher eingesetzten und weitverbreiteten Programmiersprachen und die Programmierung von Handhabungs- und Montageaufgaben durch das Teach-in-Verfahren sind unflexibel und nicht in der Lage auf Bauteil- bzw. Umgebungstoleranzen zu reagieren. Erst durch den Einsatz von Aktionsprimitiven ist eine sensorgeführte und fehlertolerante Programmierung von Robotern möglich. In diesem Programmierparadigma wird daher Potential zur Einführung der kraftgeführten Roboter montage in der Industrie gesehen.

## 1.4 Vorgehensweise

Die Entwicklung einer Kontaktregelung von Parallelrobotern auf der Basis von Aktionsprimitiven lässt sich in folgenden Schritte einteilen: Ausgangspunkt der Überlegungen ist das in Kapitel 2 beschriebene Programmierparadigma der Aktionsprimitive. Dessen Anwendung führt auf eine Steuerungsarchitektur, die den Task-Frame-Formalismus nach Bruyninckx und De Schutter [1996] integriert und die Definition so genannter Bewegungs module, die die Bewegungsbefehle der Aktionsprimitive realisieren.

Die unterlagerte Antriebsregelung kapselt den Parallelroboter und stellt eine Schnittstelle in kartesischen Koordinaten zur Verfügung über die die Bewegungs module die berechnete Solltrajektorie des Endeffektors austauschen können. Im nächsten Schritt wird daher der

Aufbau der an die Eigenschaften von Parallelrobotern angepassten unterlagerten Antriebsregelung ausführlich dargelegt. Weiterhin wird in Kapitel 3 auf den Stabilitätsbeweis der unterlagerten Antriebsregelung mit Hilfe der Passivitätstheorie eingegangen.

Die Programmierung einer Montageaufgabe mit Aktionsprimitiven führt auf eine bisher noch nicht betrachtete Darstellung der Systemdynamik im Umgebungskontakt. Diese wird im Kapitel 4 sowohl für ein statisches als auch für ein dynamisches Umgebungsmodell hergeleitet. Darüber hinaus erfordert die globale Betrachtungsweise der Kontaktregelung mit Transitionsphase die Simulation nicht-glatte mechanischer Effekte und damit die Integration differential-algebraischer Gleichungen (DAE). Diesem Umstand wird durch die Einführung von Moreaus Stoßprozess Rechnung getragen.

Damit sind alle Voraussetzungen für die Synthese einer Kontaktregelung in Kapitel 5 geschaffen. Ein struktumschaltendes Regelungskonzept (siehe Abschnitt 5.1) integriert die unterschiedlichen Regler der drei Phasen der Kontaktregelung, bevor in den Abschnitten 5.2 und 5.3 die Auslegung der Kraft-Momentenregelung für ein statisches und dynamisches Umgebungsmodell bzw. der Transitionsplaner beschrieben wird. Der Beweis in Abschnitt 5.4 verifiziert die Stabilität der Kontaktregelung in der Freiraumbewegung und während der Kontaktphase. Die Simulation mit Moreaus Stoßprozess validiert die Stabilität der Transitionsphase und schließt die Entwicklung ab.

Die in Kapitel 6 angestellten experimentellen Untersuchungen zeigen praktische Anwendungen der entwickelten Kontaktregelung auf der Basis von Aktionsprimitiven und deren Leistungsfähigkeit bevor in Kapitel 7 eine Zusammenfassung der vorliegenden Arbeit und ein Ausblick auf mögliche Weiterentwicklungen formuliert werden.

## Kapitel 2

# Robotersteuerung auf der Basis von Aktionsprimitiven

DIE in diesem Kapitel vorgestellte Robotersteuerung realisiert eine durchgängige Prozesskette von der Programmierung einer Roboter Aufgabe auf der Basis von Aktionsprimitiven bis zur Antriebsregelung eines Parallelroboters [Maaß u. a., 2006]. Auf der obersten Ebene stellt sie damit eine universelle und flexible Schnittstelle zu verschiedenen Anwendungsprogrammen, wie z. B. grafischen Programmeditoren und Montageplanern, bereit. Der zentrale Steuerungskern integriert maschinennahe Sicherheits- sowie Überwachungsfunktionen und koordiniert die einzelnen Module einer hybriden Regelung. Die unterste Ebene bildet ein Feldebussystem [Kohn u. a., 2004], das die Kommunikation zwischen den Steuerungskomponenten ermöglicht.

### 2.1 Aktionsprimitive

Die Roboterprogrammierung mit Aktionsprimitiven, erstmals vorgestellt von Hasegawa u. a. [1992] und weiterentwickelt durch die Forschergruppe um Wahl (siehe Finkemeyer u. a. [2005b]), bildet die theoretische Grundlage und ist Ausgangspunkt der Steuerungsentwicklung. Das Aktionsprimitivkonzept basiert auf dem von Mason [1981] entwickelten und von Bruyninckx und De Schutter [1996] erweiterten Compliance-Frame-Formalismus, der komplexe Roboter Aufgaben in elementare Teilaufgaben, deren Beschreibung die Aktionsprimitive darstellen, zerlegt. Innerhalb eines Aktionsprimitivs kann in einem frei wählbaren Koordinatensystem jeder Achse eindeutig eine Regelungsaufgabe zugeordnet werden. Diese wird so lange ausgeführt bis eine Abbruchbedingung gültig wird. Zur Auswertung der Abbruchbedingungen können die Zustandsgrößen des Parallelroboters oder externe Sensoren herangezogen werden. So ist eine sensorbasierte und fehlertolerante Ausführung von Montageaufgaben möglich. Das Konzept der externen Sensoren kann dabei auf beliebig komplexe Systeme erweitert werden. Über diese Schnittstelle kann z. B. ein Bildverarbeitungssystem eingebunden werden. Die folgende Zusammenfassung von Aspekten der Roboterprogrammierung mit Aktionsprimitiven basiert auf den schon erwähnten Arbeiten von Finkemeyer u. a. [2005b] und Thomas u. a. [2005].

In Abbildung 2.1(a) sind beispielhaft zwei mögliche Definitionen eines Aktionsprimitivs

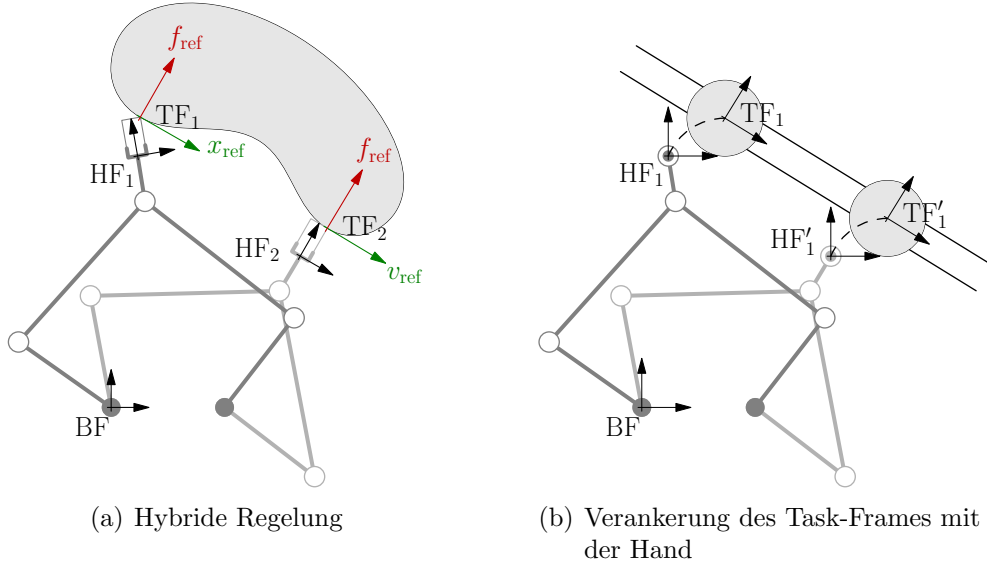


Abb. 2.1: Roboterprogrammierung mit Aktionsprimitiven

zur Konturverfolgung an einem einfachen Werkstück gezeigt. Das festzulegende Koordinatensystem wird als Task-Frame TF bezeichnet und stellt eine homogene Transformation  ${}^{\text{BF}}\mathbf{T}_{\text{TF}}$  bezogen auf ein Referenzsystem, hier das Roboterkoordinatensystem BF, dar. Der Bewegungsbefehl ist formal eine hybride Regelungsaufgabe

$$\mathcal{C} := \left\{ \langle \text{Reglertyp}, x_{\text{ref}} \rangle_i : i \in [1; n]_{\mathbb{N}} \right\}$$

die jedem Freiheitsgrad  $i$  des Parallelroboters einen Reglertyp mit entsprechendem Sollwert  $x_{\text{ref}}$  zuordnet. In beiden in Abbildung 2.1(a) gezeigten Fällen soll die Normalkraft  $f_{\text{ref}}$  auf die Werkstückoberfläche geregelt werden, während der Endeffektor parallel dazu eine bestimmte Strecke abfährt  $x_{\text{ref}}$  bzw. mit einer definierten Geschwindigkeit  $v_{\text{ref}}$  verfahren soll.

Die Einführung eines so genannten Ankers

$$\mathcal{A} \in \{\text{BF}, \text{HF}, \dots\}$$

erweitert die Möglichkeiten des Aktionsprimitivkonzepts. Diese Option ermöglicht die Definition einer virtuellen Verbindung zwischen dem Task-Frame TF und einem beliebigen Koordinatensystem in der Umgebung des Parallelroboters. Wird der Task-Frame mit dem Basiskoordinatensystem BF verbunden, bleibt seine Position und Orientierung im Verlauf der Regelung konstant. Eine Verankerung mit dem Handkoordinatensystem HF führt den Task-Frame TF kontinuierlich der aktuellen Roboterpose nach. Dadurch ist die Definition einer Bewegung relativ zur aktuellen Roboterpose und eine Ausrichtung des Endeffektors an der Umgebung möglich. Wesentlich flexibler ist allerdings die Möglichkeit den Task-Frame mit einem Sensorkoordinatensystem SF aus der Umgebung zu verankern. In Abbildung 2.1(b) ist der Task-Frame mit einer Transporteinrichtung verbunden und folgt so

während der Handhabung bzw. Montage einem bewegten Objekt.

Das Ende einer Teilaufgabe wird durch eine Abbruchbedingung  $\lambda$  bestimmt. Formal stellt diese durch Vergleichsfunktionen eine Abbildung der Roboterzustände und verfügbaren Sensorwerte auf einen boolschen Wert dar

$$\lambda = (\mathbf{x}, \mathbf{s}, t) \mapsto \{\text{true}, \text{false}\}.$$

Dabei sind  $\mathbf{x}$  die Roboterzustände wie Lage und Geschwindigkeit,  $\mathbf{s}$  die Sensorzustände und  $t$  die Ausführungszeit. Wird die Abbruchbedingung wahr, so ist das Aktionsprimitiv beendet. Eine typische Abbruchbedingung für die zweite Konturverfolgungsaufgabe aus Abbildung 2.1(a) wäre

$$\lambda = \left\{ (x \geq x_1) \vee (f > f_{\max}) \vee (t > t_{\max}) \right\}.$$

Der Parallelroboter fährt mit der vorgegebenen Geschwindigkeit und Kontaktkraft an der Kontur entlang bis er die Position  $x_1$  erreicht hat. Die zusätzliche Begrenzung der Kontaktkraft  $f$  auf  $f_{\max}$  dient der Sicherheit, während die Beschränkung der Ausführungszeit  $t$  einer Verklemmung der Steuerung vorbeugt. Die Beendigung einer Regelungsaufgabe durch eine Abbruchbedingung ist das entscheidende Element der Sensorintegration. Dadurch ist der Anwender nicht mehr gezwungen, die Umgebung des Parallelroboters exakt zu kennen und diese Informationen der Steuerung mitzuteilen. Statt dessen erfassen Sensoren die tatsächlichen Umgebungszustände und -parameter und können so flexibel auf Toleranzen der Werkstücke und des Roboters reagieren. Eine Montageaufgabe mit Parallelrobotern ohne Greifer mit künstlicher Nachgiebigkeit ist in einer realistischen Umgebung so überhaupt erst möglich.

Eine komplexe Roboteraufgabe besteht aus mehreren Aktionsprimitiven, die in der richtigen Reihenfolge und mit geeigneten Startwerten sowie Sollwerten ausgeführt werden müssen. Zu diesem Zweck werden einzelne Aktionsprimitive in speziellen Datenstrukturen, so genannten Aktionsprimitivnetzen, zusammengefasst. Diese Netze sind Graphen mit gerichteten Kanten, deren Knoten je ein Aktionsprimitiv repräsentieren. Die Kanten symbolisieren den Wechsel von einer Teilaufgabe in eine nächste, wenn die Abbruchbedingung ausgelöst hat. Über Bedingungen, die den Kanten zugeordnet sind, können Verzweigungen im Programmablauf durchgeführt werden. Diese Bedingungen bilden, ähnlich den Abbruchbedingungen, Roboterzustände oder Sensorwerte auf einen boolschen Wert ab. Gehen von einem Knoten mehrere Kanten ab, so muss sichergestellt sein, dass immer nur eine Bedingung wahr werden kann.

Die Beendigung eines Aktionsprimitivs und der Übergang zu einer neuen Teilaufgabe stellt die Regelungstechnik vor ein Stabilitätsproblem: Nach dem Auslösen einer Abbruchbedingung kann die übergeordnete Schicht im Allgemeinen das nachfolgende Aktionsprimitiv nicht in Echtzeit berechnen und die Informationen der Regelung zur Verfügung stellen. Es muss

also eine globale Vereinbarung darüber bestehen, welcher Zustand des Systems nach dem Auslösen der Abbruchbedingung und vor dem Beginn des neuen Aktionsprimitivs eingenommen werden soll. Die dafür notwendigen Berechnungen müssen zusätzlich noch in Echtzeit ausgeführt werden können. Gilt die Annahme, dass sich das System zu einem Zeitpunkt, an dem die Abbruchbedingung wahr wird, in einem sicheren Zustand befindet, kann es beliebig lange darin verharren. Daher werden am Ende eines Aktionsprimitivs die aktuellen Istwerte aller im Eingriff befindlichen Regler zu den neuen Sollwerten und das System verharrt in diesem Zustand bis ein neues Aktionsprimitiv geladen wird. Die Abarbeitung eines Aktionsprimitivs erfolgt somit in zwei Phasen: (1) Der Ausführung der Regelungsaufgabe  $\mathcal{C}$  und (2) dem Verharren in einem stabilen Zustand bis zum Eintreffen der nächsten Teilaufgabe. Die Gültigkeit dieser Annahme wird durch die Umgebung und physikalischen Grenzen des Roboters eingeschränkt. Eine Kante, an der der Roboter geschwindigkeitsgeregelt entlang fährt kann verschwinden oder der Parallelroboter fährt in eine Singularität. Der Anwender ist daher in letzter Instanz für die Vorgabe eines sicheren und zielführenden Aktionsprimitivnetzes verantwortlich. Eine ausführliche Diskussion findet sich in Finkemeyer [2004].

Umschaltende Regelungen in einer Koordinatenachse können mit Abbruchbedingungen nicht realisiert werden, da die Latenzzeiten bis zum Eintreffen einer neuen Reglerkonfiguration mit den entsprechenden Sollwerten zu groß sind. Die bisherige Definition eines Aktionsprimitivs wird daher um die Möglichkeit erweitert, in einer Koordinatenachse mehrere Reglertypen mit dazugehörigen Sollwerten festzulegen. Die mehrfachen Reglerdefinitionen werden unterschiedlichen Ebenen zugeordnet. Ist es einem Regler in einer oberen Ebene nicht möglich, seine Aufgabe auszuführen oder soll der Reglertyp aus einem anderen Grund gewechselt werden, so sendet dieser eine Nachricht an die Steuerung, die daraufhin die Umschaltung zu einem in der nachfolgenden Ebene definierten Regelalgorithmus vornimmt. Der schon bekannte Bewegungsbefehl  $\mathcal{C}$  wird daher wie folgt erweitert:

$$\mathcal{C}^* := \left\{ \langle j, \mathcal{C} \rangle : j \in [1; m]_{\mathbb{N}} \right\}.$$

Dabei ist  $j$  die Nummer der Ebene, deren Anzahl  $m$  vom Anwender vorgegeben wird. In jeder dieser Ebenen kann durch die Definition des Bewegungsbefehls  $\mathcal{C}$  allen Koordinatenachsen ein Regelziel zugeordnet werden.

Greifer und sonstige Geräte am Endeffektor oder der Peripherie des Parallelroboters können über Werkzeugkommandos  $\tau$  Befehle aus dem Roboterprogramm erhalten. Über diese Schnittstelle ist es möglich, Greifer zu öffnen bzw. zu schließen und Sensoren mit verschiedenen Datensätzen zu parametrisieren, um sie exakter an die aktuelle Aufgabenstellung anzupassen.

Die formale Definition eines Aktionsprimitivs nach Finkemeyer u. a. [2005b] lautet:

**Definition 1 (Aktionsprimitiv)** Ein Aktionsprimitiv ist definiert als

$$\mathcal{S} := \{ \mathcal{H}, \text{RF}, \mathcal{A}, \lambda, \tau, \sigma \},$$



wobei  $\mathcal{H}$  die hybride Bewegung, RF den Namen des Referenzsystems,  $\mathcal{A}$  die Verankerung des Task-Frames,  $\lambda$  die Abbruchbedingung,  $\tau$  die Werkzeugkommandos und  $\sigma := [\mathbf{x}, \mathbf{s}]^T$  die Roboterzustandsgrößen und Sensorwerte am Ende der ersten Phase eines Aktionsprimitivs darstellt.

Die hybride Bewegung

$$\mathcal{H} := \{\mathbf{T}_{\text{RF}}^{\text{RF}}, \mathcal{C}^*\}$$

ist eine Menge, bestehend aus der homogenen Transformation vom Referenzsystem RF in den Task-Frame TF und den Reglern  $\mathcal{C}^*$ . Die Bewegungsbefehle

$$\mathcal{C}^* := \left\{ \langle j, \mathcal{C} \rangle_j : j \in [1; m]_{\mathbb{N}} \right\} \quad \text{mit} \quad \mathcal{C} := \left\{ \langle \text{Reglertyp}, x_{\text{ref}} \rangle_i : i \in [1; n]_{\mathbb{N}} \right\}$$

enthalten für eine bis  $m$  Ebenen jeweils die Reglertypen für jeden Freiheitsgrad  $i$  des Parallelroboters mit dazugehörigem Sollwert.

Als Referenz RF und Anker  $\mathcal{A}$  kann jedes bekannte Koordinatensystem ausgewählt werden:

$$\text{RF}, \mathcal{A} \in \{\text{BF}, \text{HF}, \text{SF}_k, \dots\} \quad \text{mit} \quad k \in \mathbb{N}.$$

Die Abbruchbedingung

$$\lambda : (\mathbf{x}, \mathbf{s}) \mapsto \{\text{true}, \text{false}\}$$

ist eine boolsche Funktion, die die Roboterzustände  $\mathbf{x}$  und Sensorzustände  $\mathbf{s}$  auf die Werte *wahr* oder *falsch* abbildet.

Das Werkzeugkommando

$$\tau \in \{\text{„Greifer offen“}, \text{„Sensor ein“}, \dots\}$$

wählt ein oder mehrere Befehle aus und sendet diese an das entsprechende Gerät.

Die obigen Ausführungen zeigen, dass das Aktionsprimitivkonzept eine flexible Schnittstelle zur Roboterprogrammierung darstellt. Es stellt hohe Anforderungen an die Robotersteuerung, die durch kommerziell verfügbare Geräte derzeit noch nicht abgedeckt werden. Im nächsten Abschnitt wird daher eine Steuerungsarchitektur für die Roboterprogrammierung mit Aktionsprimitiven vorgestellt.

## 2.2 Steuerungsarchitektur

Die grundlegende Arbeit über eine Robotersteuerungsarchitektur auf der Basis von Aktionsprimitiven ist von Finkemeyer [2004] verfasst und darauf aufbauend eine Implementierung

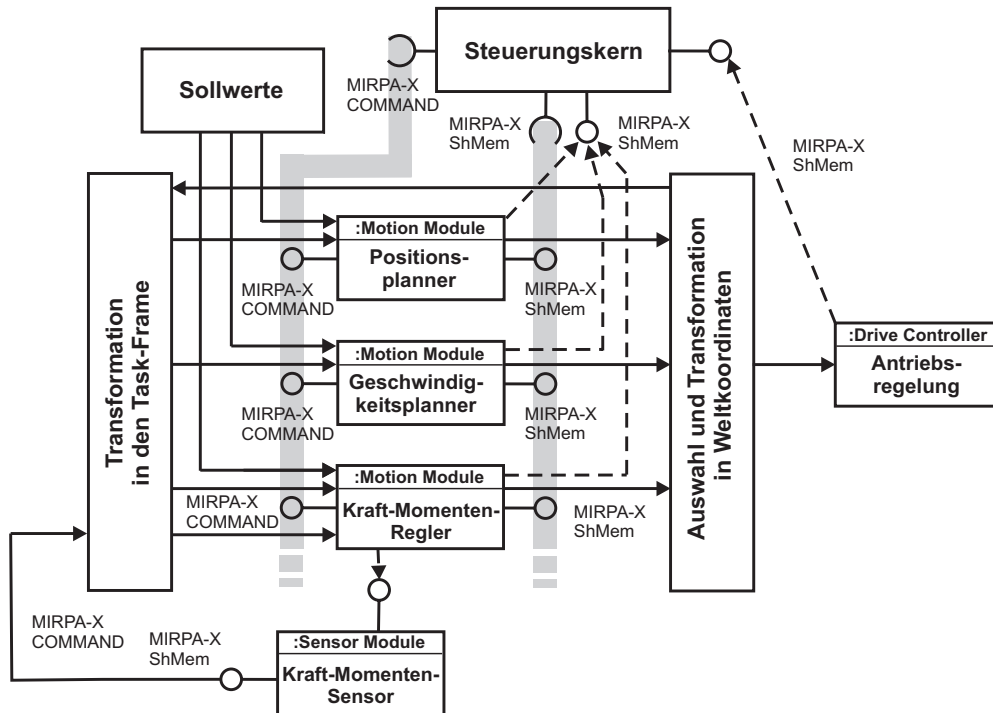


Abb. 2.2: Robotersteuerungsarchitektur RCA562

für serielle Roboter realisiert worden. Die hier besprochene Implementierung für Parallelroboter wird in [Maaß u. a., 2006; Thomas u. a., 2005] vorgestellt. Die Steuerungsarchitektur ermöglicht die Verarbeitung der Aktionsprimitive, die ein Anwender in einem Aktionsprimitivnetz zusammengestellt hat, bzw. die durch ein Montageplanungssystem automatisch erstellt wurden.

Die Funktionsweise der Steuerung kann an Hand des Signalflussbilds in Abbildung 2.2 nachvollzogen werden. Die Steuerung erhält das abzuarbeitende Aktionsprimitiv als Eingabe von einer übergeordneten Schicht, die die Abarbeitung eines Aktionsprimitivnetzes durch Berechnung der Transitionen zwischen den einzelnen Aktionsprimitiven vornimmt. Das ankommende Aktionsprimitiv gibt die Reglertypen und Sollwerte für alle Freiheitsgrade des Parallelroboters vor. Der Steuerungskern extrahiert die Informationen und überprüft in einem ersten Schritt, ob die aktuelle Konfiguration des Systems die so spezifizierte Regelungsaufgabe ausführen kann. Ist das Ergebnis dieser Prüfung positiv, werden in den einzelnen Koordinatenachsen die gewünschten Reglertypen aktiviert und mit Sollwerten versorgt. Jede Regelungsaufgabe z. B. wie die Lage- oder Kontaktregelung, wird von einem speziellen Bewegungsmodul übernommen. Bei den Bewegungsmodulen handelt es sich um erweiterte Führungsgrößengeneratoren, die im Task-Frame die Bewegung des Parallelroboters berechnen und eine Solltrajektorie generieren. Diese wird durch einen erweiterten Task-Frame-Formalismus [Maaß u. a., 2006] in das Koordinatensystem des Parallelroboters transformiert und der unterlagerten Antriebsregelung übermittelt, die schließlich die Bewegungen des Endeffektors kontrolliert. Das Ende der Ausführung eines Aktionsprimitivs wird

durch die Abbruchbedingung angezeigt und an die übergeordnete Schicht weitergeleitet, die daraufhin ein neues Aktionsprimitiv bereitstellt.

Aus der Abbildung 2.2 und der Ablaufbeschreibung über die Verarbeitung eines Aktionsprimitivs wird ersichtlich, dass die Solltrajektorie an die unterlagerte Antriebsregelung weitergeleitet wird, aber keine Rückkopplung der Roboterzustände wie Lage, Geschwindigkeit oder Beschleunigung erfolgt. Dies liegt in den Eigenschaften der Parallelroboter begründet (siehe Abschnitt 1.1.2). Da die Roboter Aufgabe im kartesischen Task-Frame programmiert wird, an den Antrieben des Parallelroboters aber nur die Antriebswinkel und Winkelgeschwindigkeiten gemessen werden können, ist die Berechnung des direkten kinematischen Problems notwendig. Bei Parallelrobotern ist dafür im Allgemeinen keine analytische Lösung gegeben, so dass eine Berechnung im Regeltakt nicht garantiert werden kann. Statt dessen kapselt die unterlagerte Antriebsregelung den Parallelroboter und garantiert über den gesamten Arbeitsraum gleichmäßiges dynamisches Verhalten, das von den Bewegungsmodulen bei der Berechnung der Solltrajektorien berücksichtigt wird. Da die Antriebsregelung in einem höheren Regeltakt ausgeführt wird als die Bewegungsmodule, kann davon ausgegangen werden, dass die tatsächlichen Roboterzustände im Task-Frame den Sollwerten entsprechen. Eine Rückführung der Roboterzustandsgrößen ist nicht erforderlich.

Einige Planungsalgorithmen der Bewegungsmodule sind auf Zustandsgrößen der Umgebung angewiesen. Spezielle Module integrieren Sensoren zur Kraft-Momentenmessung, Abstandsmessung o. ä. deren Messwerte durch eine inverse Transformation vom Sensorkoordinatensystem in den Task-Frame umgerechnet werden.

Ein weiterer wesentlicher Bestandteil der Steuerungsarchitektur sind die Überwachungsfunktionen, die integriert werden müssen, um einen sicheren Betrieb zu gewährleisten. Hier sind im Falle eines Parallelroboters besonders die Arbeitsraumüberwachung zur Laufzeit [Hesselbach u. a., 2005] und spezielle Notbremsalgorithmen zu nennen. Sie sind erforderlich, da Singularitäten im Arbeitsraum existieren und eine Bremsung der einzelnen Antriebe koordiniert erfolgen muss, um eine Zerstörung der Parallelroboterstruktur zu verhindern.

Die Kommunikation innerhalb der Steuerungsarchitektur erfolgt nachrichtenbasiert und für echtzeitkritische Daten über reservierte Speicherbereiche. Befehle vom Steuerungskern für die einzelnen Bewegungsmodule und die Daten über die Schnittstelle der Aktionsprimitive werden asynchron als Nachrichten verschickt, während die generierten Solltrajektorien zyklisch kommuniziert werden. Die Middleware MiRPA-X<sup>1</sup> [Kohn u. a., 2004] stellt diese Kommunikationsinfrastruktur mit Fehlerüberwachungs- und Scheduling-Algorithmen zur Verfügung.

## 2.2.1 Bewegungsmodule

Bewegungsmodule nehmen die vom Aktionsprimitiv vorgegebenen Planungs- bzw. Regelungsaufgaben im Task-Frame wahr. Es existieren zwei Klassen von Bewegungsmodulen:

<sup>1</sup>Abkürzung für Middleware for Robot and Process Automation – eXtended

Diejenigen, die ohne weitere Messgrößen nur aus den übergebenen Sollwerten eine Solltrajektorie berechnen und jene, die zusätzliche Sensorwerte benötigen. Zur ersten Klasse gehören Lage- und Geschwindigkeitsplaner. Sie generieren auf der Basis des aktuellen Roboterzustands eine Solltrajektorie, die das gewünschte Verhalten bewirkt. Demgegenüber beziehen Kraft-Momentenplaner, Abstandsplaner u. ä. zusätzliche Sensorwerte in ihre Berechnungen mit ein. Hierbei handelt es sich strenggenommen nicht um Planungsalgorithmen, sondern um Regler. Um eine einheitliche Nomenklatur zu wahren, wird im Zusammenhang mit Bewegungsmodulen jedoch immer von Planern gesprochen. Für jeden im Aktionsprimitiv spezifizierten Reglertyp muss ein Bewegungsmodul vorhanden sein.

Sowohl die unterlagerte Antriebsregelung als auch das Aktionsprimitivkonzept stellen Anforderungen an die Bewegungsmodule. Alle Planungsalgorithmen müssen eine Solltrajektorie

$$T_{\text{ref}} := \left\{ {}^{\text{TF}}\mathbf{x}_{\text{ref}}, {}^{\text{TF}}\dot{\mathbf{x}}_{\text{ref}}, {}^{\text{TF}}\ddot{\mathbf{x}}_{\text{ref}} \right\}$$

mit einer definierten Lage  ${}^{\text{TF}}\mathbf{x}_{\text{ref}}(t)$  sowie dem dazugehörigen Geschwindigkeits- und Beschleunigungsprofil  ${}^{\text{TF}}\dot{\mathbf{x}}_{\text{ref}}(t)$  bzw.  ${}^{\text{TF}}\ddot{\mathbf{x}}_{\text{ref}}(t)$  im Task-Frame erzeugen. Bahnplanungsalgorithmen für die Freiraumbewegung des Parallelroboters erfüllen diese Voraussetzung [Olomski, 1989; Pietsch, 2003]. Allerdings erzeugen Kraft-Momentenregler mit unterlagerter Lageregelung nur einen Lagesollwert [vgl. Sciavicco und Siciliano, 2000, Kap. 6]. Demgegenüber berücksichtigt der vorgestellte Ansatz zur Kontaktregelung von Parallelrobotern die Anforderung der Steuerungsarchitektur auf der Basis von Aktionsprimitiven und erzeugt eine vollständige Solltrajektorie.

Die Umsetzung des Aktionsprimitivkonzepts erfordert weiterhin, dass alle Planungsalgorithmen jederzeit unterbrechbar und mit beliebigen Initialwerten der Roboterzustände ausgeführt werden können, da das Auslösen einer Abbruchbedingung und der damit verbundene Wechsel der Bewegungsmodule nicht vorhersagbar ist.

Zwischen dem Ende der im aktuellen Aktionsprimitiv definierten Regelungsaufgabe durch Auslösen einer Abbruchbedingung und dem Eintreffen eines neuen Aktionsprimitivs muss ein Bewegungsmodul gemäß des oben definierten Verhaltens einen sicheren Zustand des Gesamtsystems gewährleisten. Zusätzlich muss jeder Planungsalgorithmus für einen oder mehrere Freiheitsgrade des Parallelroboters gleichzeitig einsetzbar sein.

Die maximale Belastbarkeit des Parallelroboters macht eine Ruckbegrenzung der generierten Solltrajektorie notwendig, um eine mechanische Beschädigung oder gar Zerstörung der Maschine zu vermeiden.

### 2.2.2 Task-Frame-Formalismus

Die verschiedenen Bewegungsmodule berechnen die Solltrajektorie in dem vom Aktionsprimitiv zugewiesenen Freiheitsgrad des Task-Frames. Der Task-Frame-Formalismus fasst die Sollwerte der einzelnen Bewegungsmodule zu einer vollständigen Trajektorie zusammen und

transformiert diese in das Roboterkoordinatensystem BF.

Grundlage der Algorithmen ist die im Folgenden dargelegte homogene Transformation. Die Lage  $\mathbf{x} = [\mathbf{r}^T, \boldsymbol{\varphi}^T]^T$  eines Starrkörpers im dreidimensionalen Raum wird durch den Positionsvektor  $\mathbf{r} = [x, y, z]^T$  und seine Orientierung  $\boldsymbol{\varphi} = [\varphi, \theta, \psi]^T$  beschrieben. Die Darstellung der Orientierung durch Eulerwinkel ist mehrdeutig [vgl. Sciavicco und Siciliano, 2000] und bedarf zu Beginn einer Festlegung. Der Task-Frame-Formalismus und alle darauf aufbauenden Gesetzmäßigkeiten in dieser Arbeit beruhen auf der Darstellung der Eulerwinkel in der Form *Roll-Pitch-Yaw* (RPY). Die Rotationsmatrix ist durch

$$\mathbf{R}(\boldsymbol{\varphi}) = \mathbf{R}_z(\psi)\mathbf{R}_y(\theta)\mathbf{R}_x(\varphi)$$

definiert, wobei  $\varphi$  die Rotation um die  $x$ -Achse,  $\theta$  die Rotation um die  $y$ -Achse und  $\psi$  die Rotation um die  $z$ -Achse darstellt. Der Operator

$$\boldsymbol{\varphi} = \text{RPY}(\mathbf{R}) \quad (2.1)$$

ist die Umkehrfunktion, die die zugehörigen Winkel einer gegebenen Rotationsmatrix berechnet.

Mit der oben beschriebenen Rotationsmatrix und der translatorischen Verschiebung lautet die homogene Transformationsmatrix

$$\mathbf{T}(\mathbf{x}) = \begin{bmatrix} \mathbf{R}(\boldsymbol{\varphi}) & \mathbf{r} \\ \mathbf{0}^T & 1 \end{bmatrix}.$$

Durch Indizes wird diese näher bezeichnet. Demnach definiert das Symbol  ${}^B\mathbf{T}_A$  die Transformation vom Koordinatensystem A in Koordinatensystem B. Das hochgestellte Präfix zeigt das *Bezugssystem* an, in dem die Transformation definiert ist. Die Definition der Matrix

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.2)$$

liefert durch die Operation

$$\mathbf{R} = \mathbf{D}^T \mathbf{T} \mathbf{D} \quad (2.3)$$

die Rotationsmatrix der Transformation. Weiterhin lautet die translatorische Verschiebung der Transformation

$$\mathbf{r} = \mathbf{D}^T \mathbf{T} \mathbf{t} \quad \text{mit dem Vektor} \quad \mathbf{t} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T. \quad (2.4)$$

Durch die Erweiterung der bekannten Definition des Task-Frame-Formalismus' [Bruyninckx und De Schutter, 1996; Kröger u. a., 2004] in der Arbeit von Maaß u. a. [2006], ist es möglich eine vollständige Trajektorie mit Geschwindigkeit und Beschleunigung zu transformieren. Der erweiterte Algorithmus berücksichtigt außerdem eine Besonderheit der Kontaktregelung: Befindet sich der Endeffektor des Parallelroboters im Umgebungskontakt, verändert sich die Struktur der unterlagerten Antriebsregelung. Neben den Antriebsmomenten wirken externe Kräfte und beeinflussen die Bewegungen des Parallelroboters. Um diese externen Kräfte zu kompensieren, wird in zahlreichen Veröffentlichungen eine positive Rückkopplung der gemessenen Endeffektorkräfte auf den Eingang der Regelstrecke vorgeschlagen. Beispielfhaft sei hier Sciavicco und Siciliano [2000] erwähnt, die diese Verfahren anwenden um die Freiheitsgrade einer kartesischen Impedanzregelung zu entkoppeln, oder die exakte Linearisierung des Roboters auch im Umgebungskontakt aufrecht zu erhalten. Allerdings wird schon in dem wegweisenden Aufsatz von McClamroch und Wang [1988] darauf hingewiesen, dass diese Mitkopplung der Endeffektorkräfte zu Instabilitäten führen kann. Statt dessen wird die Aufschaltung der Sollkräfte auf den Eingang der Regelstrecke empfohlen. Der Task-Frame-Formalismus wendet dieses Verfahren an, so dass neben der Solltrajektorie auch die Sollkraft vom Task-Frame TF in das Roboterkoordinatensystem BF transformiert wird.

Der Task-Frame-Formalismus definiert dazu die drei im Folgenden erläuterten Fälle:

Fall 1: Alle Freiheitsgrade des Parallelroboters sind lagegeregelt und der Task-Frame TF ist ortsfest im Roboterkoordinatensystem BF definiert: Hierbei handelt es sich um eine hochdynamische Verfahrbewegung im Arbeitsraum, die hauptsächlich während einer Handhabungsaufgabe ausgeführt wird. Ein Lageplaner berechnet aus der aktuellen Lage und dem im Aktionsprimitiv vorgegebenen Zielpunkt des Parallelroboters eine zeitoptimale und in allen Freiheitsgraden synchronisierte Trajektorie mit der Solllage  ${}^{\text{TF}}\mathbf{x}_{\text{ref}}$  im Task-Frame. Die Lage des Endeffektors im Roboterkoordinatensystem wird durch die Transformation

$${}^{\text{BF}}\mathbf{T}_{\text{HF}} = {}^{\text{BF}}\mathbf{T}_{\text{TF}}^0 \begin{bmatrix} {}^{\text{TF}}\mathbf{R}(\boldsymbol{\varphi}_{\text{ref}}) & {}^{\text{TF}}\mathbf{r}_{\text{ref}} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (2.5)$$

berechnet, so dass

$$\begin{bmatrix} \mathbf{r} \\ \boldsymbol{\varphi} \end{bmatrix} = \begin{bmatrix} \mathbf{D}^T {}^{\text{BF}}\mathbf{T}_{\text{HF}} \mathbf{t} \\ \text{RPY}(\mathbf{D}^T {}^{\text{BF}}\mathbf{T}_{\text{HF}} \mathbf{D}) \end{bmatrix}. \quad (2.6)$$

Dabei ist HF das Koordinatensystem des Endeffektors und der Index 0 bezeichnet die entsprechende Größe zu Beginn des Aktionsprimitivs. Auf eine Auflistung der Transformationen für die Geschwindigkeit und Beschleunigung wird hier verzichtet

und lediglich auf die Veröffentlichung [Maaß u. a., 2006] verwiesen.

Fall 2: Ausführung einer hybriden Regelung wobei der Task-Frame TF ortsfest im Roboterkoordinatensystem BF definiert ist. Die Lage des Endeffektors wird durch die Transformation

$${}^{\text{BF}}\mathbf{T}_{\text{HF}} = {}^{\text{BF}}\mathbf{T}_{\text{TF}}^0 \begin{bmatrix} \mathbf{I} & {}^{\text{TF}}\mathbf{r}_{\text{HF}}^0 \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} {}^{\text{TF}}\mathbf{R}(\boldsymbol{\varphi}_{\text{ref}}) & {}^{\text{TF}}\mathbf{r}_{\text{ref}} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} {}^{\text{TF}}\mathbf{R}_{\text{HF}}^0 & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (2.7)$$

berechnet. Hierbei handelt es sich um eine einfache Transformation der Solltrajektorie vom Task-Frame TF in das Roboterkoordinatensystem BF. Sie hängt von der Lage des Task-Frames und des Endeffektors zu Beginn des Aktionsprimitivs ab.

Die generalisierten Kräfte im Roboterkoordinatensystem lauten

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{HF}} {}^{\text{HF}}\mathbf{R}_{\text{TF}} {}^{\text{TF}}\boldsymbol{\zeta} \\ {}^{\text{BF}}\mathbf{R}_{\text{HF}} \left( ({}^{\text{HF}}\mathbf{r}_{\text{TF}} \times {}^{\text{HF}}\mathbf{R}_{\text{TF}} {}^{\text{TF}}\boldsymbol{\zeta}) + {}^{\text{HF}}\mathbf{R}_{\text{TF}} {}^{\text{TF}}\boldsymbol{\mu} \right) \end{bmatrix}. \quad (2.8)$$

Dabei sind  ${}^{\text{TF}}\boldsymbol{\zeta}$  die Kräfte und  ${}^{\text{TF}}\boldsymbol{\mu}$  die Momente im Task-Frame TF.

Fall 3: Ausführung einer hybriden Regelung wobei der Task-Frame mit dem beweglichen Koordinatensystem des Endeffektors HF verankert ist. Die Lage des Endeffektors wird durch die Transformation

$${}^{\text{BF}}\mathbf{T}_{\text{HF}} = {}^{\text{BF}}\mathbf{T}_{\text{TF}}^0 \begin{bmatrix} {}^{\text{TF}}\mathbf{R}(\boldsymbol{\varphi}_{\text{ref}}) & {}^{\text{TF}}\mathbf{r}_{\text{ref}} \\ \mathbf{0}^T & 1 \end{bmatrix} {}^{\text{TF}}\mathbf{T}_{\text{HF}}^0 \quad (2.9)$$

berechnet. Im Gegensatz zu Fall 2 ändert sich das Bezugssystem der Solltrajektorie im Verlauf der Regelung.

Die generalisierten Kräfte im Roboterkoordinatensystem lauten in diesem Fall

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}} {}^{\text{TF}}\boldsymbol{\zeta} \\ {}^{\text{BF}}\mathbf{R}_{\text{TF}} {}^{\text{TF}}\boldsymbol{\mu} \end{bmatrix}. \quad (2.10)$$

Die während eines Umgebungskontakts am Endeffektor auftretenden Kräfte werden zur Trajektorienplanung der Kontaktregelung in dem entsprechenden Bewegungsmodul benötigt. In Abhängigkeit der definierten Regelungsaufgabe werden sie vom Roboterkoordinatensystem BF in den Task-Frame TF zurücktransformiert:

Fall 2: Die generalisierten Kräfte in einem ortsfest definierten Task-Frame TF lauten

$${}^{\text{TF}}\mathbf{f} = \begin{bmatrix} {}^{\text{TF}}\mathbf{R}_{\text{BF}} {}^{\text{BF}}\boldsymbol{\zeta} \\ {}^{\text{TF}}\mathbf{R}_{\text{BF}} {}^{\text{BF}}\boldsymbol{\mu} \end{bmatrix}. \quad (2.11)$$

Fall 3: Die generalisierten Kräfte in einem variablen Task-Frame TF lauten

$${}^{\text{TF}}\mathbf{f} = \begin{bmatrix} {}^{\text{TF}}\mathbf{R}_{\text{BF}} {}^{\text{BF}}\boldsymbol{\zeta} \\ {}^{\text{TF}}\mathbf{R}_{\text{BF}} \left( {}^{\text{BF}}\boldsymbol{\mu} + {}^{\text{BF}}\mathbf{r} \times {}^{\text{BF}}\boldsymbol{\zeta} \right) \end{bmatrix}. \quad (2.12)$$

## 2.3 Antriebstechnik

Die unterlagerte Antriebsregelung erhält vom Steuerungskern mit den Bewegungsmodulen und dem Task-Frame-Formalismus die Solltrajektorie im Roboterkoordinatensystem und berechnet daraus die Stellgrößen der Antriebe in Gelenkkoordinaten.

Die beschriebene Robotersteuerungsarchitektur ist für den Einsatz von Direktantrieben konzipiert. Abhängig von der Parallelstruktur werden für den Antrieb rotatorischer Gelenke leistungsstarke Servomotoren oder eisenlose Linearmotoren für prismatische Gelenke verwendet, um die Vorteile von Parallelrobotern bei hochdynamischen Verfahrbewegungen auszunutzen. Die Vorteile von Direktantrieben gegenüber Getriebemotoren liegen unter anderem in der verminderten Reibung und der geringeren Elastizität. Ohne ein vorgeschaltetes Getriebe werden außerdem mechanisches Spiel und Rückschlageffekte vermieden. Dafür muss in der dynamischen Modellierung der Parallelroboterstruktur das Antriebsträgheitsmoment integriert werden, da der reduzierende Faktor der Getriebeübersetzung wegfällt. Änderungen der Massenträgheitsmatrix, bedingt durch Lageänderungen des Parallelroboters, oder Nutzlaständerungen am Endeffektor wirken sich wesentlich direkter auf die Dynamik aus und werden im Regelungskonzept berücksichtigt.

In den Umrichtern zur Ansteuerung der Antriebe ist eine Strom- und Geschwindigkeitsregelung integriert. Sie stellen nach außen eine Schnittstelle zur Verfügung, um die vom unterlagerten Antriebsregler berechneten Vorsteuergrößen in Gelenkkoordinaten einspeisen zu können und liefern die Istwerte der Antriebswinkel  $\mathbf{q}$  und Geschwindigkeiten  $\dot{\mathbf{q}}$  in digitaler Form zurück.

Die Kommunikation zwischen den Steuerungskomponenten, d.h. zwischen dem Steuerungskern auf dem PC-basierten Echtzeitbetriebssystem QNX<sup>2</sup> und den Umrichtern findet über ein Feldbussystem mit dem *Industrial Automation Protocol* (IAP) [Kohn u. a., 2004] statt.

---

<sup>2</sup>QNX ist ein eingetragenes Warenzeichen der Firma QNX Software Systems Ltd.



## Kapitel 3

### Unterlagerte Antriebsregelung

DIE unterlagerte Antriebsregelung stellt der Steuerungsarchitektur eine kartesische Schnittstelle zur Verfügung. In diese kann die von den Bewegungsmodulen generierte Solltrajektorie eingespeist werden. Die Regelstrecke, bestehend aus Umrichtern, Antrieben und dem Parallelroboter, wird als stabiles Teilsystem mit gleichmäßiger Dynamik gekapselt, so dass eine bahntreue und ausreichend schnelle Folgeregung entlang der vorgegebenen Solltrajektorie sichergestellt. Die unterlagerte Antriebsregelung realisiert das hohe Potenzial paralleler Strukturen für schnelle und genaue Verfahrbewegungen und stellt damit die Voraussetzung der Kontaktregelung dar.

Es werden mathematische und mechanische Eigenschaften von Parallelrobotern bei der Entwicklung eines geeigneten Regelungskonzepts in Betracht gezogen: Die Lage des Endeffektors wird in Gelenkkoordinaten geregelt, deren Messung direkt an den Ausgängen der Umrichter möglich ist. Die Berechnung des direkten kinematischen Problems ist dagegen nicht notwendig. Der Aufbau einer Kaskadenregelung mit innerem Geschwindigkeitsregelkreis erlaubt die Berücksichtigung der Umrichterzeitkonstante in der Reglerauslegung. Mit der Methode der berechneten Momente<sup>1</sup> zur Ermittlung geeigneter Vorsteuergrößen wird das dynamische Modell in die unterlagerte Antriebsregelung integriert. So wird der ausgeprägten Lageabhängigkeit der Massenträgheitsmatrix und der Matrix der Coriolis- und Zentrifugalkräfte bei Parallelrobotern Rechnung getragen sowie die kartesische Schnittstelle implementiert. Eine zusätzliche Verzögerung der Vorsteuergrößen und die Anpassung der Parameter des Geschwindigkeitsreglers zur Laufzeit erhöht die Bahntreue der Folgeregung.

Im Folgenden werden die einzelnen Komponenten der unterlagerten Antriebsregelung beschrieben. Die Methoden der Reglerauslegung werden ausführlich dargelegt. In den beiden letzten Abschnitten wird die Stabilität mit Hilfe der Passivitätstheorie nachgewiesen und das Verhalten der Regelung diskutiert.

#### 3.1 Kaskadenregelung

Die Kaskadenregelung des Parallelroboters besteht aus einem inneren Geschwindigkeitsregelkreis und der äußeren Lageregung und wurde in [Leonhard, 2000] vorgestellt. Das

---

<sup>1</sup>In der englischsprachigen Literatur wird dieser Ansatz als *computed-torque feed forward algorithm* bezeichnet



modelliert. Die Zeitkonstante der Stromregelung  $T_c$  ist in der Diagonalmatrix  $\mathbf{K}_{T_c} = T_c \cdot \mathbf{I}$  mit  $\mathbf{I} \in \mathbb{R}^{n \times n}$  enthalten. Die Eingangsgröße der Umrichter wird durch den Geschwindigkeitsregler  $\mathbf{u}_1$  und die Vorsteuerung  $\boldsymbol{\tau}_{\text{fwd}}$  vorgegeben.

Die Messung der Antriebswinkel im Umrichter wird im Regeltakt dem übergeordneten Lageregler digital übergeben. Das Rauschen dieses Messsignals ist zu vernachlässigen und es treten keine Verzögerungen auf. Im Gegensatz dazu wird das Geschwindigkeitssignal durch numerische Differenzierung gewonnen oder durch einen Beobachter im Umrichter geschätzt. Die auftretenden Verzögerungen werden in dem Verfahren zur Reglersynthese berücksichtigt und sind im Blockschaltbild durch ein  $\text{PT}_1$ -Element mit der Zeitkonstanten  $T_m$  nachgebildet

$$\mathbf{K}_{T_m} \ddot{\mathbf{q}}_m + \dot{\mathbf{q}}_m = \dot{\mathbf{q}}. \quad (3.4)$$

Dabei ist  $\mathbf{K}_{T_m} = T_m \cdot \mathbf{I}$  und  $\mathbf{I} \in \mathbb{R}^{n \times n}$ . In obiger Gleichung ist  $\dot{\mathbf{q}}$  die aus (3.1) bekannte tatsächliche und  $\dot{\mathbf{q}}_m$  die verzögerte Geschwindigkeit.

Für die Geschwindigkeitsregelung kommt ein PI-Regler zum Einsatz, dessen Verstärkungsfaktor in Abhängigkeit von der Roboterpose verändert werden kann. Im äußeren Lageregelkreis ist der Einsatz verschiedener Reglertypen möglich. In diesem Fall ist aus weiter unten beschriebenen Gründen ein  $\text{PT}_1\text{D}$ -Regler im Eingriff.

### 3.1.1 Geschwindigkeitsregelung

Der PI-Geschwindigkeitsregler für den  $k$ -ten Antrieb mit dem Verstärkungsfaktor  $V_1^{(k)}$  und der Zeitkonstanten  $T_{i,1}^{(k)}$  hat die Übertragungsfunktion

$$G_{K_1}^{(k)}(s) := \frac{U_1(s)}{E_1(s)} = V_1^{(k)} \frac{T_{i,1}^{(k)} s + 1}{T_{i,1}^{(k)} s}, \quad k \in [1; n]_{\mathbb{N}} \quad (3.5a)$$

während im Mehrgrößenfall die Darstellung

$$\Sigma_{K_1} := \begin{cases} \dot{\mathbf{x}}_1 = \mathbf{K}_{V_1} \mathbf{K}_{T_{i,1}}^{-1} \mathbf{e}_1 \\ \mathbf{u}_1 = \mathbf{x}_1 + \mathbf{K}_{V_1} \mathbf{e}_1 \end{cases} \quad \text{mit} \quad \mathbf{e}_1 = \mathbf{u}_2 + \dot{\mathbf{q}}_{\text{fwd}} - \dot{\mathbf{q}}_m \quad (3.5b)$$

zweckmäßiger ist. Die Regeldifferenz  $\mathbf{e}_1$  als Eingangsgröße des PI-Geschwindigkeitsreglers wird aus der Stellgröße des Lagereglers  $\mathbf{u}_2$ , der Vorsteuergröße  $\dot{\mathbf{q}}_{\text{fwd}}$  und der Regelgröße  $\dot{\mathbf{q}}_m$  selbst gebildet. In den Matrizen  $\mathbf{K}_{V_1} = \text{diag}(V_1^{(1)}, \dots, V_1^{(n)})$  und  $\mathbf{K}_{T_{i,1}} = \text{diag}(T_{i,1}^{(1)}, \dots, T_{i,1}^{(n)})$  sind die Reglerparameter zusammengefasst.

Eine Adaption des Trägheitsmoments ist möglich, da der Verstärkungsfaktor  $V_1^{(k)}$  ausschließlich von der Roboterpose abhängig ist. Über ein zu definierendes Adaptionsgesetz wird der PI-Regler zur Laufzeit der aktuellen Massenträgheitsmatrix  $\mathbf{M}(\mathbf{q})$  angepasst. Besonders in den Randbereichen des Arbeitsraums lässt sich dadurch eine signifikante Verbesserung

der Regelgüte erzielen.

### 3.1.2 Lageregelung

Nachdem der innere Geschwindigkeitsregelkreis den Parallelroboter als nichtlineare Strecke stabilisiert und Störungen stationär genau ausregelt, führt ein äußerer Kreis die Antriebswinkeldifferenz auf einen Lageregler. Aufgrund der überlegenen Störsprungantwort, die durch Brunotte [2001] nachgewiesen wurde, wird ein PT<sub>1</sub>D-Lageregler eingesetzt. Dessen Übertragungsfunktion für den  $k$ -ten Antrieb des Parallelroboters lautet

$$G_{K_2}^{(k)}(s) := \frac{U_2(s)}{E_2(s)} = V_2^{(k)} \frac{T_{r,2}^{(k)} s + 1}{T_{l,2}^{(k)} s + 1}, \quad k \in [1; n]_{\mathbb{N}} \quad \text{mit} \quad T_{l,2} > T_{r,2}. \quad (3.6a)$$

Für den Mehrgrößenfall gilt wiederum die Darstellung im Zustandsraum

$$\Sigma_{K_2} := \begin{cases} \dot{\mathbf{x}}_2 = -\mathbf{K}_{T_{l,2}}^{-1} \mathbf{x}_2 + \mathbf{K}_{T_{l,2}}^{-1} \mathbf{e}_2 \\ \mathbf{u}_2 = \mathbf{K}_{V_2} \left( \mathbf{I} - \mathbf{K}_{T_{r,2}} \mathbf{K}_{T_{l,2}}^{-1} \right) \mathbf{x}_2 + \mathbf{K}_{V_2} \mathbf{K}_{T_{r,2}} \mathbf{K}_{T_{l,2}}^{-1} \mathbf{e}_2 \end{cases} \quad (3.6b)$$

mit  $\mathbf{e}_2 = \mathbf{q}_{\text{fwd}} - \mathbf{q}$ . Die Eingangsgröße  $\mathbf{e}_2$  ist die Differenz aus der Vorsteuergröße  $\mathbf{q}_{\text{fwd}}$  und den Antriebswinkeln  $\mathbf{q}$ , während die Reglerparameter in den Matrizen  $\mathbf{K}_{V_2} = \text{diag}(V_2^{(1)}, \dots, V_2^{(n)})$ ,  $\mathbf{K}_{T_{l,2}} = \text{diag}(T_{l,2}^{(1)}, \dots, T_{l,2}^{(n)})$  und  $\mathbf{K}_{T_{r,2}} = \text{diag}(T_{r,2}^{(1)}, \dots, T_{r,2}^{(n)})$  zusammengefasst sind.

### 3.1.3 Auslegung der Kaskadenregelung

Ausgehend von der Bewegungsgleichung (3.1) wird die unterlagerte Antriebsregelung in einer Kaskadenstruktur aufgebaut. Das angewandte Auslegeverfahren zur Minimierung der Regelfläche wurde in [Brunotte, 2001] eingeführt und wird im Folgenden exemplarisch für einen Antrieb  $k \in [1; n]_{\mathbb{N}}$  dargestellt. Die Ergebnisse lassen sich analog auf alle weiteren Antriebe anwenden. Das ursprüngliche Verfahren wurde für einen einschleifigen Regelkreis entwickelt. In dieser Arbeit wird die Stabilitätsuntersuchung für die Anwendung mit einem Parallelroboter als nichtlineare Mehrgrößenstrecke erweitert. Da das Führungsverhalten der unterlagerten Antriebsregelung von einer modellbasierten Vorsteuerung bestimmt wird, können die Regler zur Optimierung des Störverhaltens ausgelegt werden. Die folgenden Normierungen

$$\begin{aligned} s &= \frac{x}{T_c} & V_2^{(k)} &= \frac{1}{cT_c} \\ V_1^{(k)} &= \frac{\bar{j}(\mathbf{q})}{bT_c} & T_{r,2}^{(k)} &= dT_c \\ T_{i,1}^{(k)} &= aT_c & T_{l,2}^{(k)} &= aT_c \end{aligned}$$

vereinfachen die komplexen mathematischen Ausdrücke. In den obigen Gleichungen sind die Parameter  $a$ ,  $b$ ,  $c$  und  $d$  sämtlich reelle Zahlen. Das effektiv auf einen Antrieb wirkende Trägheitsmoment wird mit  $\bar{j}$  benannt und ist eine Funktion der Antriebswinkel  $\mathbf{q}$ .

Im weiteren Verlauf der Reglerauslegung kann eine Nullstelle des Geschwindigkeitsreglers durch einen Pol des PT<sub>1</sub>D-Lagereglers kompensiert werden. Die Übertragungsfunktionen der Regler lauten dann

$$G_{K_1}^{(k)}(x) = \frac{1}{abx} \frac{\bar{j}(\mathbf{q})}{T_c} \quad \text{und} \quad (3.7a)$$

$$G_{K_2}^{(k)}(x) = \frac{dx + 1}{c} \frac{1}{T_c}. \quad (3.7b)$$

Die Störübertragungsfunktion von einem unbekannten Lastmoment zum Systemausgang ist

$$F_{ga}^{(k)} = \frac{abc(x+1)x}{abc \cdot x^4 + abc \cdot x^3 + ac \cdot x^2 + (c+d)x + 1} \cdot \frac{T_c^2}{\bar{j}(\mathbf{q})} \quad (3.8)$$

und die dazugehörige Regelfläche ergibt sich zu

$$A_R = abc \frac{T_c^2}{\bar{j}(\mathbf{q})}. \quad (3.9)$$

Eine Minimierung der Regelfläche  $A_R$  unter den durch einen Koeffizientenvergleich mit einem Vergleichspolynom vorgegebenen Randbedingungen liefert die Reglerparameter

$$V_1^{(k)} = \frac{5D^2 + 1}{16D^2} \frac{\bar{j}(\mathbf{q})}{T_c}, \quad V_2^{(k)} = \frac{1}{4(1 + 2D^2) T_c}, \quad (3.10a)$$

$$T_{i,1}^{(k)} = \frac{4T_c(5D^2 + 1)}{1 + 2D^2}, \quad T_{r,2}^{(k)} = 4T_c, \quad (3.10b)$$

$$T_{l,2}^{(k)} = T_{i,1}^{(k)}. \quad (3.10c)$$

Die Dämpfung  $D$  der Kaskadenregelung ist der verbleibende Entwurfparameter. Für die unterlagerte Antriebsregelung wird ein kritisch gedämpftes Systemverhalten angestrebt. Mit der so vorgegebenen Dämpfung von  $D = 1$  berechnen sich die Reglerparameter wie folgt:

$$V_1^{(k)} = \frac{3}{8} \frac{\bar{j}(\mathbf{q})}{T_c}, \quad V_2^{(k)} = \frac{1}{12} \frac{1}{T_c}, \quad (3.11a)$$

$$T_{i,1}^{(k)} = 8T_c, \quad T_{r,2}^{(k)} = 4T_c, \quad (3.11b)$$

$$T_{l,2}^{(k)} = T_{i,1}^{(k)}. \quad (3.11c)$$

Eine besondere Stellung nimmt die Verstärkung des Geschwindigkeitsreglers in der vordeuten Gleichung (3.11a) ein.  $V_1^{(k)}$  ist abhängig von der Lage des Endeffektors im Arbeitsraum und wird während des Betriebs nachgeführt.

Regelung	ITAE in $10^{-6} \text{ rad} \cdot \text{s}$	
	Antrieb 1	Antrieb 2
konstante Verstärkung	574,4	127,6
lageabhängige Verstärkung	444,0	103,5
Reduktion der Regelfläche	22,7%	18,9%

Tab. 3.1: Zeitgewichtete Betragsregelfläche der Geschwindigkeitsregelung

### 3.1.4 Anpassung der Parameter des Geschwindigkeitsreglers

Die Beschreibung der Wirkungsweise in einem *einschleifigen* Geschwindigkeitsregelkreis verdeutlicht das Prinzip der Anpassung des Parameters der Geschwindigkeitsregelung. Gemäß der Auslegungsvorschrift in Gleichung (3.11) ist die Verstärkung des Geschwindigkeitsreglers proportional zum Trägheitsmoment  $j$  des Antriebs:

$$V_1 = \frac{5D^2 + 1}{16D^2} \frac{j(q)}{T_c} \propto j(q). \quad (3.12)$$

Der Verstärkungsfaktor wird mit einem neutralen Trägheitsmoment von  $j_0 := 1$  ausgelegt. Während des Betriebs wird die Stellgröße mit dem aktuellen Trägheitsmoment  $j(q)$  multipliziert, so dass der Geschwindigkeitsregler der Lage des Parallelroboters angepasst ist. Dieses Verfahren führt auf ein gleichförmiges Systemverhalten im gesamten Arbeitsraum und wird in der Literatur z. B. von Leonhard [2000] beschrieben.

Im Fall des hier gegebenen MIMO-Systems kann, wie in Abschnitt 1.1.2 gezeigt, weder die Lageabhängigkeit noch die Verkopplung der Massenträgheitsmatrix  $\mathbf{M}(\mathbf{q})$  vernachlässigt werden. Durch die Einbeziehung der gesamten Massenträgheitsmatrix

$$\hat{\mathbf{u}}_1 = \hat{\mathbf{M}}(\mathbf{q}_{\text{ref}}) \mathbf{u}_1 \quad (3.13)$$

wird der Stellgrößenvektor  $\mathbf{u}_1$  des Geschwindigkeitsreglers (3.5) mit der aus den Sollwerten der Antriebswinkel  $\mathbf{q}_{\text{ref}}$  berechneten Massenträgheitsmatrix des Robotermodells  $\hat{\mathbf{M}}$  multipliziert.

Durch die Anpassung der Reglerverstärkung verbessert sich die Störsprungantwort deutlich. Beispielfhaft werden in Abbildung 3.2 die simulierten Störsprungantworten für den Parallelroboter HEXA in einer Pose am Rande des Arbeitsraums gezeigt. Ein Qualitätsvergleich der Regelung mit und ohne Anpassung wurde über die zeitgewichtete Betragsregelfläche ITAE vorgenommen und in Tabelle 3.1 zusammengefasst. Die Störgröße wirkte ausschließlich auf den ersten Antrieb, so dass in Abbildung 3.2(a) der Einfluss auf die tatsächlich gestörte Achse und in Abbildung 3.2(b) die Auswirkungen der Verkopplungen im MIMO-System beobachtet werden kann. Demnach verringert sich die ITAE der gestörten Achse um 22,7% und die Verkopplungen um 18,9%.

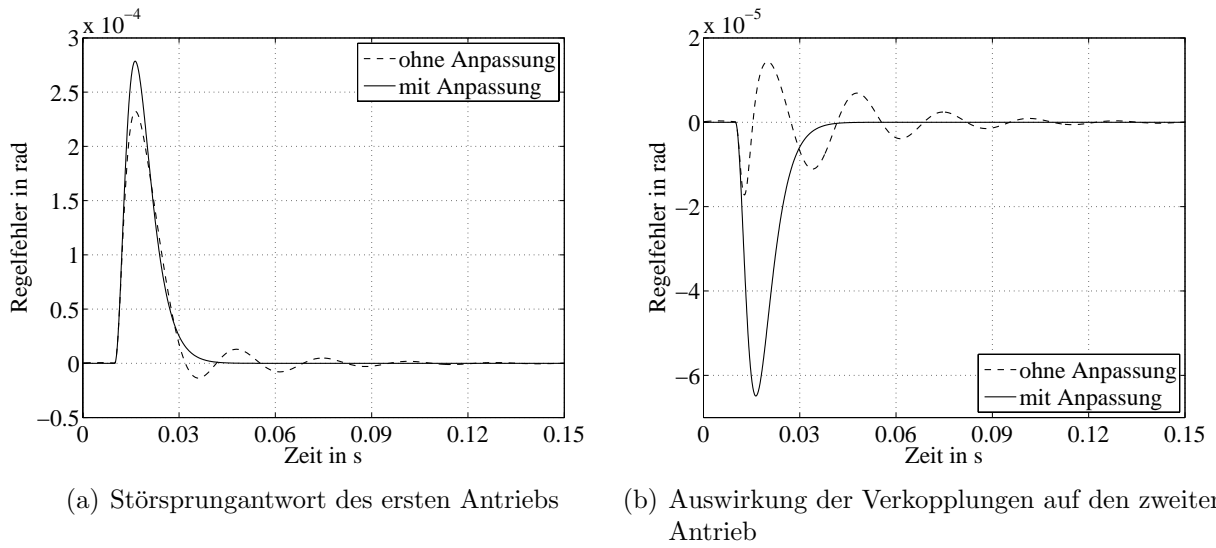


Abb. 3.2: Störsprungantwort der Geschwindigkeitsregelung für eine Roboterpose am Rande des Arbeitsraums. Die Störgröße wirkt auf Antrieb 1.

## 3.2 Vorsteuerung

Das inverse dynamische Modell des Parallelroboters wird zur Ermittlung der Vorsteuergrößen durch die Methode der berechneten Momente in die unterlagerte Antriebsregelung integriert. Im Allgemeinen liegt das Modell in kartesischen Koordinaten vor [vgl. Tsai, 1999, Kap. 10], so dass die berechneten Antriebskräfte durch die inverse transponierte Jacobimatrix  $\mathbf{J}_x^{-T}$  in Gelenkkordinaten transformiert werden müssen:

$$\boldsymbol{\tau}_{\text{fwd}} = \mathbf{J}_x^{-T} \left[ \hat{\mathbf{M}}(\mathbf{x}_{\text{ref}}) \ddot{\mathbf{x}}_{\text{ref}} + \hat{\mathbf{C}}(\mathbf{x}_{\text{ref}}, \dot{\mathbf{x}}_{\text{ref}}) \dot{\mathbf{x}}_{\text{ref}} + \hat{\mathbf{g}}(\mathbf{x}_{\text{ref}}) \right]. \quad (3.14)$$

In obiger Gleichung ist  $\hat{\mathbf{M}}$  die berechnete Massenträgheitsmatrix, während  $\hat{\mathbf{C}}$  die Matrix der berechneten Zentrifugal- bzw. Corioliskräfte darstellt und  $\hat{\mathbf{g}}$  die berechneten Gewichtskräfte symbolisiert. Das resultierende Vorsteuermoment  $\boldsymbol{\tau}_{\text{fwd}}$  wird aus der Solltrajektorie berechnet, die durch den Task-Frame-Formalismus in das Roboterkoordinatensystem BF transformiert wurde und die Sollbeschleunigung  $\ddot{\mathbf{x}}_{\text{ref}}$ , die Sollgeschwindigkeit  $\dot{\mathbf{x}}_{\text{ref}}$  und die Sollage  $\mathbf{x}_{\text{ref}}$  definiert. Fehler in der Vorsteuerung, bedingt durch Modellungenauigkeiten, treten nur im Vorwärtspfad der Regelung auf. Die vollständige unterlagerte Antriebsregelung, bestehend aus der Kaskadenregelung mit Parallelroboter und Vorsteuerung, ist in Abbildung 3.3 dargestellt.

Die Sollwerte des Lage- bzw. Geschwindigkeitsreglers lassen sich mit den Gleichungen

$$\dot{\mathbf{q}}_{\text{ref}} = \mathbf{J}_x \dot{\mathbf{x}}_{\text{ref}} \quad (3.15)$$

$$\mathbf{q}_{\text{ref}} = \mathbf{n}(\mathbf{x}_{\text{ref}}) \quad (3.16)$$





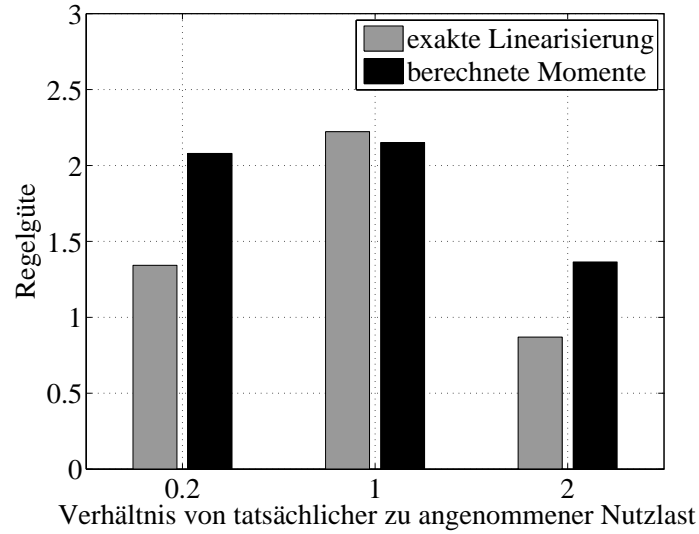


Abb. 3.4: Vergleich der Regelgüte (invertierte Betragsregelfläche) von modellbasierten Regelungskonzepten bei variierendem Verhältnis der tatsächlichen zur angenommenen Nutzlast ( $m_l/\hat{m}_l$ )

mit Parallelrobotern auf die Bahntreue ankommt, während eine Verzögerung in der Ausführung toleriert werden kann, ist diese Maßnahme anzuwenden. Die Forderung zur Auslegung der Führungsgrößenfilter für den  $k$ -ten Antrieb im Bildbereich lautet

$$s^2 q^{(k)} = G_c s^2 q_{\text{fwd}}^{(k)} \Rightarrow q^{(k)} = G_c q_{\text{fwd}}^{(k)}. \quad (3.17)$$

Dabei symbolisiert  $G_c$  die Übertragungsfunktion des Umrichters. Alle Signale gemäß Abbildung 3.3 eingesetzt, folgt daraus

$$s^2 q^{(k)} = G_c \left[ s^2 q_{\text{fwd}}^{(k)} + G_{K_1} \left( s q_{\text{fwd}}^{(k)} (G_2 - G_m G_c) + G_{K_2} q_{\text{fwd}}^{(k)} (G_1 - G_c) \right) \right] \quad (3.18)$$

mit der Übertragungsfunktion der Messverzögerung des Geschwindigkeitssignals  $G_m$ . Damit die Forderung (3.17) im stationären Fall erfüllt wird, muss der Vorfilter des Geschwindigkeitssollwerts die Übertragungsfunktion

$$G_2 := G_m G_c = \frac{1}{(T_m s + 1)(T_c s + 1)}. \quad (3.19)$$

und der Vorfilter des Lagesollwertes die Übertragungsfunktion

$$G_1 := G_c = \frac{1}{T_c s + 1} \quad (3.20)$$

besitzen. Da die Umrichterverzögerung  $T_c$  und die Messverzögerung  $T_m$  für alle Antriebe

gleich sind, lautet die Darstellung der Führungsgrößenfilter für alle Antriebe des Parallelroboters

$$\mathbf{G}_1(s) := \left\{ \frac{1}{T_c s + 1} \right\}_n \quad \text{und} \quad \mathbf{G}_2(s) := \left\{ \frac{1}{(T_m s + 1)(T_c s + 1)} \right\}_n. \quad (3.21)$$

Der Index  $n \in \mathbb{N}$  symbolisiert dabei die Anzahl der Freiheitsgrade.

Die Zustandsgrößen des Parallelroboters Lage und Geschwindigkeit werden, wie in Abbildung 3.3 zu sehen, nicht an die übergeordnete Steuerungsarchitektur zurückgeführt. Statt dessen wird davon ausgegangen, dass die unterlagerte Antriebsregelung den, an der kartesischen Schnittstelle in einem geringeren Takt übergebenen Sollwerten, folgen kann. Voraussetzung für diese Kapselung ist die asymptotische Stabilität der unterlagerten Antriebsregelung, die im folgenden Abschnitt bewiesen wird.

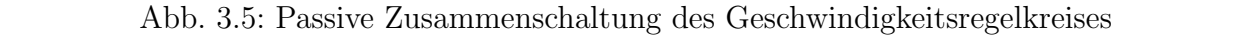
### 3.3 Stabilität der Antriebsregelung

Im Fall der unterlagerten Antriebsregelung des Parallelroboters handelt es sich um ein nicht-lineares Mehrgrößensystem. Die Reglerauslegung gemäß Abschnitt 3.1.3 führt daher nicht mehr auf ein zwangsläufig stabiles System und muss statt dessen gesondert nachgewiesen werden. Um die Verzögerung durch die Umrichter in den Stabilitätsbeweis integrieren zu können, wird ein Vorgehen mit den Methoden der Passivitätstheorie [Khalil, 2002; van der Schaft, 2000] angewandt. Damit unterscheidet sich dieser Ansatz in wesentlichen Aspekten von den in [Cheah u. a., 2003; Siciliano und Villani, 1999] u. a. vorgebrachten Stabilitätsbeweisen.

Die unterlagerte Antriebsregelung wird in passive Teilsysteme aufgespalten, deren Zusammenschaltung wiederum passiv ist. Gemäß den Theoremen der Passivitätstheorie, die im Anhang B zusammengefasst sind, werden so Kandidaten für eine Ljapunow-Funktion des Systems ermittelt. Anschließend wird die asymptotische Stabilität der Ruhelagen mit dem Invarianzprinzip von Krasowskij-LaSalle [vgl. dazu Slotine und Li, 1991] gezeigt. So kann die Stabilität des inneren Geschwindigkeitsregelkreises und darauf aufbauend die Stabilität der gesamten Reglerkaskade nachgewiesen werden.

#### 3.3.1 Passivität

Der Geschwindigkeitsregelkreis der unterlagerten Antriebsregelung kann als passive Zusammenschaltung von drei im Folgenden beschriebenen, wiederum passiven, Teilsystemen angesehen werden. Die Verzögerung durch die Messsysteme wird bei diesem Vorgehen vernachlässigt. Das Ersatzschaltbild der notwendigen Umformungen ist in Abbildung 3.5 dargestellt. Der Parallelroboter mit der Bewegungsgleichung (3.1) ist ein Euler-Lagrange-System, das nach Ortega u. a. [1998] einen passiven Operator von den Antriebsmomenten zu den Ge-


$$\Sigma_a : \tau \mapsto \dot{\mathbf{q}} \quad (3.22)$$
$$\mathcal{H}_a = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_q \dot{\mathbf{q}} + \mathcal{V}_g(\mathbf{q}) \quad (3.23)$$

Die Reihenschaltung des PI-Geschwindigkeitsreglers (3.5) und der Umrichterverzögerung für einen Antrieb wird durch die Übertragungsfunktion

beschrieben. Sie besitzt eine relative Ordnung von eins. Daher ist es möglich sie durch die Partialbruchzerlegung

in zwei Systeme erster Ordnung aufzuspalten, so dass die Gesetze der Passivitätstheorie angewendet werden können. Zusammen mit dem passiven Operator des Parallelroboters (3.22) bilden sie, wie in Abbildung 3.5 gezeigt, eine passive Zusammenschaltung passiver Teilsys-

teme.

Die Differentialgleichung

$$\mathbf{K}_{T_c} \dot{\mathbf{y}} + \mathbf{y} = -\mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \dot{\mathbf{q}} \quad (3.26)$$

beschreibt das Teilsystem im ersten Term der Gleichung (3.25). Es setzt sich aus der Umrichterverzögerung sowie einem zusätzlichen Verstärkungsfaktor zusammen und definiert den passiven Operator

$$\Sigma_b : -\dot{\mathbf{q}} \mapsto \mathbf{y} \quad (3.27)$$

von der Geschwindigkeit  $\dot{\mathbf{q}}$  zur Zwischengröße  $\mathbf{y}$ . Die Speicherfunktion

$$\mathcal{H}_b = \frac{1}{2} \mathbf{y}^T \mathbf{K}_{T_c} \mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \mathbf{y} \quad (3.28)$$

enthält die Matrizen  $\mathbf{K}_{T_c} = T_c \cdot \mathbf{I}$  und  $\mathbf{K}_{V_1} = \text{diag}(V_1^{(1)}, \dots, V_1^{(n)})$ . In Matrix  $\mathbf{K}_{T_c}$  stellt der Parameter  $T_c$  die Zeitkonstante der Umrichterverzögerung dar.  $V_1^{(k)}$  ist der Verstärkungsfaktor des Geschwindigkeitsreglers für die  $n$  Antriebe des Parallelroboters. Schließlich repräsentiert  $\mathbf{I}$  die Einheitsmatrix. Im Folgenden wird der zusätzliche Verstärkungsfaktor der Umrichterverzögerung in diesem Zweig des Blockschaltbilds durch

$$\mathbf{K}_b := \mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \quad (3.29)$$

abgekürzt. Die kompakte Notation der Speicherfunktion lautet somit

$$\mathcal{H}_b = \frac{1}{2} \mathbf{y}^T [\mathbf{K}_{T_c} \mathbf{K}_b] \mathbf{y}. \quad (3.30)$$

Der zweite Term in Gleichung (3.25), bestehend aus dem I-Anteil sowie der Verstärkung des Geschwindigkeitsreglers, wird durch die Differentialgleichung

$$\dot{\mathbf{z}} = -\mathbf{K}_{T_i}^{-1} \mathbf{K}_{V_1} \dot{\mathbf{q}} \quad (3.31)$$

beschrieben und definiert den passiven Operator

$$\Sigma_c : -\dot{\mathbf{q}} \mapsto \mathbf{z} \quad (3.32)$$

ebenfalls von der Geschwindigkeit  $\dot{\mathbf{q}}$  zur einer weiteren Zwischengröße  $\mathbf{z}$ . Die zugehörige Speicherfunktion ist

$$\mathcal{H}_c = \frac{1}{2} \mathbf{z}^T \left[ \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \right] \mathbf{z} \quad (3.33)$$

mit  $\mathbf{K}_{T_{i,1}} = \text{diag}(T_{i,1}^{(1)}, \dots, T_{i,1}^{(n)})$ .

In einer asymptotisch stabilen Ruhelage der Lageregelung des Parallelroboters ist die Sollgeschwindigkeit, die in das hier untersuchte Teilsystem eingespeist wird, immer null. Daher bilden die beiden Systeme  $\Sigma_b$  und  $\Sigma_c$  jeweils eine negative Rückführung des passiven Operators  $\boldsymbol{\tau} \mapsto \dot{\mathbf{q}}$  des Parallelroboters. Unter diesen Bedingungen lässt sich die gemeinsame Versorgungsrate der Geschwindigkeitsregelung aus Abbildung 3.5 wie folgt herleiten:

$$\dot{\mathbf{q}}^T \mathbf{e} = \dot{\mathbf{q}}^T (\mathbf{z} + \mathbf{d}) = \dot{\mathbf{q}}^T (\mathbf{z} + \mathbf{y} + \boldsymbol{\tau}) = \dot{\mathbf{q}}^T \boldsymbol{\tau} + \mathbf{y}^T \dot{\mathbf{q}} + \mathbf{z}^T \dot{\mathbf{q}}. \quad (3.34)$$

Diese setzt sich wiederum aus den Versorgungsraten der Teilsysteme  $\Sigma_a$  bis  $\Sigma_c$  zusammen. Aus diesem Grund ist das gesamte System passiv [vgl. Khalil, 2002]. Die Gesamtspeicherfunktion ist die Summe aller Teilspeicherfunktionen

$$\begin{aligned} \mathcal{H} &= \mathcal{H}_a + \mathcal{H}_b + \mathcal{H}_c \\ &= \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_q \dot{\mathbf{q}} + \mathcal{V}_g(\mathbf{q}) + \frac{1}{2} \mathbf{y}^T [\mathbf{K}_{T_c} \mathbf{K}_b] \mathbf{y} + \frac{1}{2} \mathbf{z}^T [\mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1}] \mathbf{z}. \end{aligned} \quad (3.35)$$

Der Integrator im Teilsystem  $\Sigma_c$  (3.31) ist ein verlustloses System und *nicht* streng ausgangspassiv. Es kann damit nicht direkt auf die Stabilität des Geschwindigkeitsregelkreises geschlossen werden. Statt dessen stellt die hergeleitete Gesamtspeicherfunktion (3.35) einen Kandidaten für eine Ljapunow-Funktion dar.

### 3.3.2 Ruhelagen des Systems

Im weiteren Verlauf des Stabilitätsbeweises nach Ljapunow werden in diesem Abschnitt die Ruhelagen des geschwindigkeitsgeregelten Parallelroboters bestimmt: Ausgehend von den Bewegungsgleichungen (3.1) und der Regelung in Gleichung (3.26) sowie (3.31) lautet das Differentialgleichungssystem des geschlossenen Kreises

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{y} + \mathbf{z} \quad (3.36a)$$

$$\mathbf{K}_{T_c} \dot{\mathbf{y}} + \mathbf{y} = -\mathbf{K}_b \dot{\mathbf{q}} \quad (3.36b)$$

$$\dot{\mathbf{z}} = -\mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \dot{\mathbf{q}}. \quad (3.36c)$$

Das Einsetzen der Gleichung (3.36b) sowie die Integration und das Einsetzen von (3.36c) in (3.36a) führt auf die Differentialgleichung des geschlossenen Regelkreises

$$\begin{aligned} \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) &= -\mathbf{K}_b \dot{\mathbf{q}} - \mathbf{K}_{T_c} \dot{\mathbf{y}} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \int_{t_0}^t \dot{\mathbf{q}} \, d\tau \\ &= -\underbrace{\mathbf{K}_b \dot{\mathbf{q}} - \mathbf{K}_{T_c} \dot{\mathbf{y}}}_{\Sigma_b} - \underbrace{\mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} [\Delta \mathbf{q} + \mathbf{q}_0]}_{\Sigma_c}. \end{aligned} \quad (3.37)$$

Im eingeschwungenen Zustand für  $t \rightarrow \infty$  liefern die Terme der abgeleiteten Größen, hier

zusammengefasst als System  $\Sigma_b$ , keinen Beitrag. Die Umrichterverzögerung mit zugehörigem Proportionalanteil ist bei verschwindender Regelabweichung null. Demgegenüber kompensiert der I-Anteil des Geschwindigkeitsreglersystems  $\Sigma_c$  die Gravitationskräfte und hält den Parallelroboter in einer konstanten Lage. Dieses Verhalten führt auf das Kräftegleichgewicht

$$\mathbf{g}(\mathbf{q}^*) = -\mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \mathbf{q}^* \quad (3.38)$$

in der Ruhelage

$$\mathbf{q}^* = \Delta \mathbf{q} + \mathbf{q}_0. \quad (3.39)$$

In obiger Gleichung ist  $\mathbf{q}_0$  die Lage des Parallelroboters zu Beginn der Geschwindigkeitsregelung und  $\Delta \mathbf{q}$  die Strecke, die bis zum Stillstand zurückgelegt wird. Für unterschiedliche Anfangsbedingungen ergibt sich demzufolge eine differente Ruhelage. Die ursprüngliche Bewegungsgleichung (3.1) wird in die Gleichgewichtslage  $\mathbf{q}^*$  des Parallelroboters verschoben, die sich im Stillstand einstellt

$$\mathbf{M}(\bar{\mathbf{q}})\ddot{\bar{\mathbf{q}}} + \mathbf{C}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}})\dot{\bar{\mathbf{q}}} + \mathbf{g}(\bar{\mathbf{q}}) = \boldsymbol{\tau} \quad \text{mit} \quad \bar{\mathbf{q}} = \mathbf{q} - \mathbf{q}^*. \quad (3.40)$$

Die globale Stabilität der Geschwindigkeitsregelung ist dann gegeben, wenn es sich bei Gleichung (3.38) um eine asymptotisch stabile Ruhelage der Bewegungsgleichung (3.40) handelt.

### 3.3.3 Ljapunow-Funktion

Eine skalare Funktion aller Zustandsgrößen eines Systems wird dann und nur dann eine *Ljapunow-Funktion* genannt, wenn sie selbst positiv semidefinit und ihre Ableitung nach der Zeit negativ definit ist. In Übereinstimmung mit den Aussagen der Passivitätstheorie ist die Speicherfunktion  $\mathcal{H}$  des Gesamtsystems aus Gleichung (3.35) ein Kandidat für eine Ljapunow-Funktion des geschwindigkeitsgeregten Parallelroboters. Die Speicherfunktion wiederum enthält Anteile kinetischer sowie potentieller Energie:

$$\mathcal{H} = \mathcal{T} + \mathcal{V},$$

wobei  $\mathcal{T}$  die kinetische und  $\mathcal{V}$  die potentielle Energie repräsentiert.

Die notwendige Bedingung für asymptotische Stabilität ist die positive Semidefinitheit der Gesamtspeicherfunktion  $\mathcal{H}$  in Gleichung (3.35). Der Anteil der kinetischen Energie

$$\mathcal{T} = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_{\bar{\mathbf{q}}} \dot{\mathbf{q}} + \frac{1}{2} \mathbf{y}^T \mathbf{K}_{T_c} \mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \mathbf{y} \quad (3.41)$$

ist positiv, wenn die Integratorzeitkonstante des PI-Geschwindigkeitsreglers jedes Antriebs

größer als die Zeitkonstante des korrespondierenden Umrichters ist:

$$T_{i,1}^{(k)} \geq T_c^{(k)} : \quad \forall k \in [1; n]_{\mathbb{N}} \quad (3.42)$$

An dieser Stelle sei darauf hingewiesen, dass es sich bei den Matrizen der Reglerparameter und der Umrichterzeitkonstante ausnahmslos um Diagonalmatrizen handelt, die diese Vereinfachungen gestatten.

Die Gewichtskraft  $\mathbf{g}(\bar{\mathbf{q}})$  und der I-Anteil des Geschwindigkeitsreglers tragen zur potentiellen Energie bei:

$$\mathcal{V} = \frac{1}{2} \mathbf{z}^T \left[ \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \right] \mathbf{z} + \mathcal{V}_g(\bar{\mathbf{q}}). \quad (3.43)$$

Der erste Term in obiger Gleichung ist immer größer als null, da sowohl die Matrix  $\mathbf{K}_{T_i}^{-1}$  als auch  $\mathbf{K}_{V_1}$  positiv definit ist. Die folgenden Betrachtungen konzentrieren sich damit auf den zweiten Term  $\mathcal{V}_g(\bar{\mathbf{q}})$ : Die potentielle Energie des geschwindigkeitsgeregelten Parallelroboters ist immer größer oder gleich der potentiellen Energie in der natürlichen Ruhelage  $\mathbf{q}^{**}$ . Diese stellt sich dann ein, wenn nur die Gravitation auf die Struktur wirkt und die Antriebe stromlos sind und bildet somit eine untere Grenze der potentiellen Energie mit

$$\mathcal{V}_g(\bar{\mathbf{q}}) \geq \mathcal{V}_g(\mathbf{q}^{**}) \in \mathbb{R}. \quad (3.44)$$

Daraus folgt, dass die potentielle Energie der natürlichen Ruhelage  $\mathcal{V}_g(\mathbf{q}^{**})$  immer kleiner als die potentielle Energie in einer beliebigen Ruhelage  $\mathbf{q}^*$  ist. Die positive Semidefinitheit der Speicherfunktion  $\mathcal{H}$  kann daher nicht für alle Anfangsbedingungen garantiert werden. Die Gleichung (3.35) wird deshalb um den konstanten Wert

$$\mathcal{V}_o = \mathcal{V}_g(\mathbf{q}^*) - \mathcal{V}_g(\mathbf{q}^{**}) \quad (3.45)$$

erweitert, so dass die potentielle Energie im gesamten Arbeitsraum immer größer als null ist. Die Speicherfunktion mit der Erweiterung um die Konstante (3.45)

$$\begin{aligned} \mathcal{H}_e = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_{\bar{\mathbf{q}}} \dot{\mathbf{q}} + \frac{1}{2} \mathbf{y}^T \mathbf{K}_{T_c} \mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \mathbf{y} \\ + \frac{1}{2} \mathbf{z}^T \left[ \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \right] \mathbf{z} + \mathcal{V}_g(\bar{\mathbf{q}}) + \mathcal{V}_o. \end{aligned} \quad (3.46)$$

stellt damit eine mögliche Ljapunow-Funktion der Geschwindigkeitsregelung dar.

In einem Vorgriff auf die im nächsten Abschnitt dargestellte Verifikation der negativen Semidefinitheit der zeitlichen Ableitung der Ljapunow-Funktion wird die Speicherfunktion nochmals modifiziert. Ohne Einschränkung der bisherigen Schlussfolgerungen wird dem

integralen Term der potentiellen Energie aus Gleichung (3.43) der Faktor  $1/\alpha^2$  hinzugefügt:

$$\begin{aligned} \mathcal{H}_e = & \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}_{\bar{\mathbf{q}}} \dot{\mathbf{q}} + \frac{1}{2} \mathbf{y}^T \mathbf{K}_{T_c} \mathbf{K}_{V_1} \left[ \mathbf{I} - \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{T_c} \right] \mathbf{y} \\ & + \frac{1}{2} \mathbf{z}^T \left[ \frac{1}{\alpha^2} \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} \right] \mathbf{z} + \mathcal{V}_g(\bar{\mathbf{q}}) + \mathcal{V}_o \end{aligned} \quad (3.47)$$

mit

$$\alpha \in \mathbb{R} \quad \text{und} \quad \alpha > 0. \quad (3.48)$$

Die Gleichung (3.47) erfüllt weiterhin die notwendigen Bedingungen einer Ljapunow-Funktion, so dass im nächsten Abschnitt der Stabilitätsbeweis fortgesetzt werden kann.

### 3.3.4 Ableitung der Ljapunow-Funktion

In diesem Abschnitt wird die zeitliche Ableitung der Ljapunow-Funktion  $\mathcal{H}_e$  aus Gleichung (3.47) entlang der Systemtrajektorien  $\mathbf{f}(\mathbf{x})$  mittels der Lie-Ableitung

$$\dot{\mathcal{H}}_e = \frac{\partial \mathcal{H}_e}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial \mathcal{H}_e}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) = L_{\mathbf{f}(\mathbf{x})} \mathcal{H}_e(\mathbf{x}) \quad (3.49)$$

berechnet. Die negative Semidefinitheit der Ableitung ist eine hinreichende Bedingung des Invarianzprinzips von Krasowskij-LaSalle zum Beweis der asymptotischen Stabilität der Ruhelage  $\mathbf{q}^*$ . Im Fall der hier untersuchten Geschwindigkeitsregelung ist die Lie-Ableitung der Konstante  $\mathcal{V}_o$  gleich null, so dass sich die Ableitung gemäß

$$\dot{\mathcal{H}}_e = L_{\mathbf{f}} \mathcal{H}_a + L_{\mathbf{f}} \mathcal{H}_b + L_{\mathbf{f}} \mathcal{H}_c \quad (3.50)$$

vereinfacht.

Die Zustandsraumdarstellung des Geschwindigkeitsregelkreises in Abbildung 3.5, bestehend aus den Bewegungsgleichungen des Parallelroboters (3.1), dem Geschwindigkeitsregler selbst sowie der Umrichterverzögerung in den Gleichungen (3.26) und (3.31), lautet

$$\begin{bmatrix} \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \\ \dot{\mathbf{y}} \\ \dot{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{M}_{\bar{\mathbf{q}}}^{-1} \mathbf{C}_{\bar{\mathbf{q}}} & -\mathbf{M}_{\bar{\mathbf{q}}}^{-1} & -\mathbf{M}_{\bar{\mathbf{q}}}^{-1} \\ \mathbf{0} & \mathbf{K}_{V_1} \left[ \mathbf{K}_{T_{i,1}}^{-1} - \mathbf{K}_{T_c}^{-1} \right] & -\mathbf{K}_{T_c}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{T_{i,1}}^{-1} \mathbf{K}_{V_1} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{q}} \\ \dot{\mathbf{q}} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}_{\bar{\mathbf{q}}}^{-1} \mathbf{g}(\bar{\mathbf{q}}) \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.51)$$

Der Zustandsvektor wird in dieser Form als  $\mathbf{x} = [\bar{\mathbf{q}}^T, \dot{\mathbf{q}}^T, \mathbf{y}^T, \mathbf{z}^T]^T$  zusammengefasst. Die Ableitung der Ljapunow-Funktion berechnet sich nach (3.50) als Produkt der Ableitung von



$\mathcal{H}_e$  nach dem Zustandsvektor und der zeitlichen Ableitung des Systemzustands

$$\dot{\mathcal{H}}_e = \begin{bmatrix} \mathbf{g}^T(\bar{\mathbf{q}}) & (\mathbf{M}_{\bar{q}}\dot{\mathbf{q}})^T & (\mathbf{K}_{T_c}\mathbf{K}_b\mathbf{y})^T & \left(\frac{1}{\alpha^2}\mathbf{K}_{T_{i,1}}^{-1}\mathbf{K}_{V_1}\mathbf{z}\right)^T \end{bmatrix} \cdot \begin{bmatrix} \dot{\mathbf{q}} \\ -\mathbf{M}_{\bar{q}}^{-1}[\mathbf{g}(\bar{\mathbf{q}}) + \mathbf{C}_{\bar{q}}\dot{\mathbf{q}} + \mathbf{y} + \mathbf{z}] \\ \mathbf{K}_{V_1} \left[ \mathbf{K}_{T_{i,1}}^{-1} - \mathbf{K}_{T_c}^{-1} \right] \dot{\mathbf{q}} - \mathbf{K}_{T_c}^{-1}\mathbf{y} \\ \mathbf{K}_{T_{i,1}}^{-1}\mathbf{K}_{V_1}\dot{\mathbf{q}} \end{bmatrix} \quad (3.52)$$

und kann schließlich als

$$\begin{aligned} \dot{\mathcal{H}}_e = & -\dot{\mathbf{q}}^T \mathbf{C}_{\bar{q}} \dot{\mathbf{q}} - \mathbf{y}^T \mathbf{K}_b \mathbf{y} + \mathbf{y}^T \left[ \mathbf{K}_{T_{i,1}}^{-2} \mathbf{K}_{V_1}^2 [\mathbf{K}_{T_{i,1}} - \mathbf{K}_{T_c}]^2 \right] \dot{\mathbf{q}} \\ & + \mathbf{z}^T \left[ \frac{1}{\alpha^2} \mathbf{K}_{T_{i,1}}^{-2} \mathbf{K}_{V_1}^2 - \mathbf{I} \right] \dot{\mathbf{q}} \end{aligned} \quad (3.53)$$

vereinfacht werden.

Aussagen über die Definitheit der obigen skalaren Funktion von mehrdimensionalen Vektoren und Matrizen sind nur bedingt möglich. Daher wird die obere Grenze der Ableitung (3.53) durch Normen der beschriebenen Systeme und Signale abgeschätzt. Durch dieses Vorgehen verringert sich die Genauigkeit der ermittelten Stabilitätsgrenzen. Die komplexen Ausdrücke der Geschwindigkeitsregelung lassen sich jedoch nur so beherrschen. Gemäß den Eigenschaften eines jeden Euler-Lagrange-Systems [vgl. Ortega u. a., 1998] gilt für die 2-Norm des Parallelroboters die Beziehung

$$\|\mathbf{C}(\bar{\mathbf{q}}, \dot{\mathbf{q}})\| \leq k_c \|\dot{\mathbf{q}}\|. \quad (3.54)$$

Die jeweiligen 2-Normen der Reglermatrizen, bei denen es sich ausnahmslos um Diagonalmatrizen handelt, sind

$$\begin{aligned} \|\mathbf{K}_{V_1}\| &= V_{1\max}, & \|\mathbf{K}_{T_{i,1}}\| &= T_{i,1\max} \quad \text{und} \\ \|\mathbf{K}_{T_c}\| &= T_c. \end{aligned}$$

Im Falle eines Vektors repräsentiert  $\|\cdot\|$  die euklidische Norm des Signals. Unter Berücksichtigung dieser Bedingungen ergibt sich die obere Grenze der zeitlichen Ableitung der

Ljapunow-Funktion zu

$$\begin{aligned} \dot{\mathcal{H}}_e \leq & -k_c \|\dot{\mathbf{q}}\|^3 - V_{1\max} \left(1 - \frac{T_C}{T_{i,1\max}}\right) \|\mathbf{y}\|^2 \\ & + \left(\frac{V_{1\max}}{T_{i,1\max}}\right)^2 (T_{i,1\max} - T_c)^2 \|\mathbf{y}\| \|\dot{\mathbf{q}}\| + \left(\left(\frac{V_{1\max}}{\alpha T_{i,1\max}}\right)^2 - 1\right) \|\mathbf{z}\| \|\dot{\mathbf{q}}\|. \end{aligned} \quad (3.55)$$

Bei der Zwischengröße  $\mathbf{z}$  handelt es sich um das Ausgangssignal eines Integrators, das über alle Grenzen wachsen kann. Demgegenüber kann die Verzögerung erster Ordnung das Geschwindigkeitssignal  $\dot{\mathbf{q}}$  maximal um den vorgeschalteten Faktor verstärken, so dass für das Signal  $\mathbf{y}$  folgende Begrenzung

$$\|\mathbf{y}\| \leq \frac{V_{1\max}}{T_{i,1\max}} (T_{i,1\max} - T_c) \|\dot{\mathbf{q}}\| \quad (3.56)$$

gilt. Das Einsetzen der obigen Ungleichung in (3.55) vereinfacht diese zu

$$\dot{\mathcal{H}}_e \leq -k_c \|\dot{\mathbf{q}}\|^3 + \left(\left(\frac{V_{1\max}}{\alpha T_{i,1\max}}\right)^2 - 1\right) \|\mathbf{z}\| \|\dot{\mathbf{q}}\|. \quad (3.57)$$

Damit die Ableitung der Ljapunow-Funktion, unabhängig vom aktuellen Systemzustand, d. h. für beliebige Signale  $\mathbf{z}$  und  $\dot{\mathbf{q}} \in \mathbb{R}^n$  negativ semidefinit ist, muss die Bedingung

$$V_{1\max} \leq \alpha \cdot T_{i,1\max} \quad \text{mit} \quad \alpha \in \mathbb{R} \quad \text{und} \quad \alpha > 0 \quad (3.58)$$

eingehalten werden.

### 3.3.5 Stabilität der Geschwindigkeitsregelung

Nachdem in den vorhergehenden Abschnitten die Bedingungen der positiven Semidefinitheit der Ljapunow-Funktion und die negative Semidefinitheit der zeitlichen Ableitung hergeleitet wurde, folgt die asymptotische Stabilität der Geschwindigkeitsregelung aus dem Invarianzprinzip von Krasowskij-LaSalle [siehe Slotine und Li, 1991, Kap. 3.4.3]: Das System in (3.51), bestehend aus Parallelroboter, Umrichter und PI-Geschwindigkeitsregler, konvergiert in die größte invariante Menge für die der Zustandsvektor zu null wird. Diese Menge entspricht der Ruhelage (3.39) des geschwindigkeitsgeregten Parallelroboters.

Anhand der Ljapunow-Funktion in Gleichung (3.47) und deren zeitlicher Ableitung (3.55) ist zu sehen, dass die Stabilität der Geschwindigkeitsregelung von den Parametern des PI-Reglers und der Verzögerung der Umrichter abhängig ist. Die in den Ungleichungen (3.42) und (3.58) formulierten Bedingungen machen deutlich, dass das System je schneller gegen die Ruhelage strebt desto größer die Integrationszeitkonstante  $T_{i,1}$  ist. Eine große Zeitkonstante verringert den Einfluss des integralen Anteils der Stellgröße und führt schließlich auf einen

reinen P-Regler. Dieser wäre deshalb sehr robust, könnte aber unberücksichtigte Effekte, wie die Reibung in den Gelenken der Parallelroboterstruktur, nicht ausregeln und würde so zu einer bleibenden Regelabweichung beim realen System führen.

Der Parameter  $k_c$  ist eine Eigenschaft des Euler-Lagrange-Systems und macht, neben der Inhomogenität der Massenträgheitsmatrix, einen wesentlichen Unterschied zwischen Parallelrobotern und deren seriellen Gegenstücken aus. Wie an der zeitlichen Ableitung der Ljapunow-Funktion in Gleichung (3.55) zu sehen, repräsentiert  $k_c$  die Dämpfung des mechanischen Systems durch die Coriolis-Matrix. Je größer dieser Faktor ist, desto weniger beeinflussen die Reglerparameter das Verhalten der Geschwindigkeitsregelung und desto *gutmütiger* ist die Regelstrecke. In einem Vergleich wurden die Werte des Parameters  $k_c$  für den Parallelroboter HEXA und den seriellen Roboter COMAU SMART ermittelt. Während die Dämpfung des Systems durch die Coriolis-Matrix für den seriellen Roboter bei  $k_{c,ser} = 0,3202$  lag, erreichte der Faktor für den Parallelroboter nur den Wert  $k_{c,par} = 0,028$ . Die beiden Werte unterscheiden sich um eine Größenordnung und verdeutlichen, dass ein Parallelroboter höhere Ansprüche an die Regelung stellt.

Der Stabilitätsbeweis der PI-Geschwindigkeitsregelung wird in der folgenden Schlussfolgerung zusammengefasst:

**Korollar 1 (Stabilität der PI-Geschwindigkeitsregelung)** *Der innere PI-Geschwindigkeitsregelkreis der unterlagerten Antriebsregelung*

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{K}_{V_1} \mathbf{K}_{T_{i,1}}^{-1} \mathbf{e}_1 \\ \mathbf{u}_1 = \mathbf{x}_1 + \mathbf{K}_{V_1} \mathbf{e}_1 \end{cases} \quad \text{mit} \quad \mathbf{e}_1 = -\dot{\mathbf{q}},$$

d. h. die Geschwindigkeitssollwerte  $\dot{\mathbf{q}}_{\text{fwd}}$  sind null, stabilisiert einen beliebigen Parallelroboter

$$\mathbf{M}(\bar{\mathbf{q}})\ddot{\mathbf{q}} + \mathbf{C}(\bar{\mathbf{q}}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\bar{\mathbf{q}}) = \boldsymbol{\tau}$$

mit der Umrichterverzögerung

$$\mathbf{K}_{T_c} \dot{\boldsymbol{\tau}} + \boldsymbol{\tau} = \mathbf{u}_1,$$

und dem beliebigen Anfangszustand  $\mathbf{x}$  aus Gleichung (3.51) wenn

a) die Integrationszeitkonstante jedes Antriebs größer oder mindestens gleich der Zeitkonstante des entsprechenden Umrichters ist

$$T_{i,1}^{(k)} \geq T_c^{(k)} : \quad \forall k \in [1; n]_{\mathbb{N}}.$$

b) die maximale Verstärkung des PI-Geschwindigkeitsreglers nicht über alle Grenzen wächst

$$V_{1\max} \leq \alpha \cdot T_{i,1\max} \quad \text{mit} \quad \alpha \in \mathbb{R} \quad \text{und} \quad \alpha > 0.$$

Dann konvergiert der Zustandsvektor  $\mathbf{x}$  für  $t \rightarrow \infty$  in die Ruhelage  $\mathbf{q}^*$ , in der sich der Parallelroboter im Stillstand befindet.

**Beispiel 1 (Stabilitätsbeweis der PI-Geschwindigkeitsregelung)** Die Auslegung des Geschwindigkeitsreglers geschieht mit den in Gleichung (3.10) definierten Gesetzen. Diese vereinfachen sich bei einer gewünschten Dämpfung der unterlagerten Antriebsregelung von  $D = 1$  zu den in Gleichung (3.11) dargestellten Formeln. Bei einer angenommenen Masse am Endeffektor von  $m_l = 3 \text{ kg}$  ergibt sich aus dem dynamischen Modell des Parallelroboters HEXA (vgl. Abschnitt A.1) ein effektives Massenträgheitsmoment zwischen  $\bar{J}_{\min} = 0,03 \text{ kgm}^2$  und  $\bar{J}_{\max} = 0,21 \text{ kgm}^2$ , je nachdem ob sich der Endeffektor am Rand oder im Zentrum des Arbeitsraums befindet. Die Umrichterzeitkonstante sei  $T_c = 600 \mu\text{s}$ , während die Verzögerung der Messsysteme vernachlässigt wird.

Mit diesen Vorgaben wird mittels der Adaption des Trägheitsmoments gemäß Gleichung (3.13) eine Verstärkung des PI-Geschwindigkeitsreglers im Intervall  $V_1 = [18,75; 131,25]$  und eine Integrationszeitkonstante desselben von  $T_{i,1} = 8 \cdot T_c = 4,8 \text{ ms}$  berechnet.

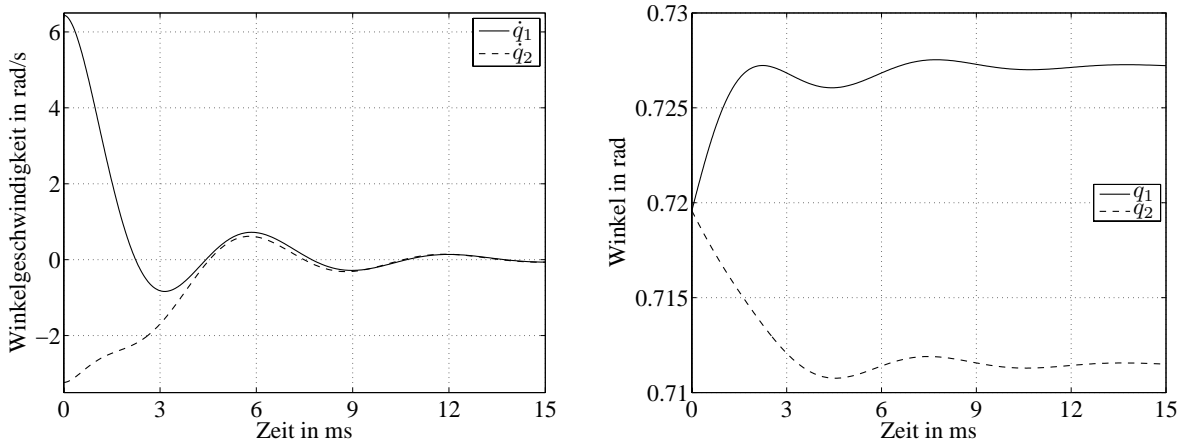
Da die Integrationszeitkonstante mit  $T_{i,1} = 8 \cdot T_c$  größer als die Umrichterzeitkonstante ist und der Verstärkungsfaktor einen maximalen Wert von  $V_{1,\max} = 131,25$  besitzt, ist die Stabilität der PI-Geschwindigkeitsregelung für jeden Punkt im Arbeitsraum des Parallelroboters gewährleistet.

Abbildung 3.6 zeigt den Verlauf der Systemzustände Winkelgeschwindigkeit  $\dot{\mathbf{q}}$  und Winkel  $\mathbf{q}$  einer Simulation der PI-Geschwindigkeitsregelung nach dem Blockschaltbild 3.5 mit dem Modell des Parallelroboters HEXA nach [Rose, 2004]. Der Endeffektor befand sich zu Beginn der Simulation im Zentrum des Arbeitsraums. Die Drehzahlsollwerte aller Antriebe betrugen  $\dot{q}_{\text{ref}} = 0 \text{ rad/s}$  und die Reglerverstärkung war konstant auf  $V_1 = 131,25$  eingestellt. In Abbildung 3.6(a) ist die Systemantwort für die Winkelgeschwindigkeiten der ersten beiden Antriebe dargestellt. Ausgehend von den unterschiedlichen Anfangsbedingungen ist deutlich der asymptotisch stabile Einschwingvorgang zu erkennen. Das korrespondierende Verhalten der Antriebswinkel ist in Abbildung 3.6(b) zu sehen. Von einer gemeinsamen Ausgangslage im Zentrum des Arbeitsraums bewegen sie sich entsprechend der eingprägten Anfangsgeschwindigkeit nach einer kurzen Transitionsphase in ihre Ruhelage und verharren dort.

### 3.3.6 Stabilität der Lageregelung

Die  $PT_1D$ -Lageregelung schließt den Regelkreis um die asymptotisch stabile Geschwindigkeitsregelung. Eine Annäherung des dynamischen Verhaltens der Geschwindigkeitsregelung durch ein  $PT_1$ -Element mit Ersatzzeitkonstante erlaubt den folgenden Stabilitätsbeweis der Lageregelung:

Ein nichtlineares System ist stabil, wenn ein Abklingkoeffizient  $\lambda$  existiert [siehe Slotine und Li, 1991, Kapitel 3.2]. Dieser entspricht dem Kehrwert der Ersatzzeitkonstante  $T_e =$



(a) Winkelgeschwindigkeit der Antriebe Nr. 1 und 2 mit unterschiedlichen Anfangsbedingungen (b) Zeitlicher Verlauf der Winkel der geschwindigkeitsgeregelten Antriebe Nr. 1 und 2

Abb. 3.6: Einschwingvorgang der Geschwindigkeitsregelung mit unterschiedlichen Anfangsbedingungen und Drehzahl Sollwert 0 rad/s. Beispielhaft sind hier die Zustandsgrößen der ersten beiden Antriebe des Parallelroboters HEXA dargestellt.

$1/\lambda$  und führt so auf eine Annäherung der Dynamik des Geschwindigkeitsregelkreises. In Abbildung 3.7(a) ist dieser Zusammenhang veranschaulicht. Die durchgehende Linie zeigt den Verlauf der 2-Norm des Zustandsvektors, als ein Maß der im System gespeicherten Energie, über den gesamten Einschwingvorgang. Die gestrichelt gezeichnete Hüllkurve der Form

$$E = \alpha \|\mathbf{x}(0)\| e^{-\frac{t}{T_{e,1}}} \quad (3.59)$$

ist immer größer als die Energie des Zustandsvektors. Demzufolge klingen die Transienten der Geschwindigkeitsregelung mit der Ersatzzeitkonstanten  $T_{e,1}$  ab.

Der  $PT_1D$ -Lageregler mit dem  $PT_1$ -Element des angenäherten Geschwindigkeitsregelkreises ist in Abbildung 3.7(c) dargestellt. Der Stabilitätsbeweis kann mit den Methoden der linearen Regelungstechnik erfolgen und ist im folgenden Korollar zusammengefasst:

**Korollar 2 (Stabilität der  $PT_1D$ -Lageregelung)** *Der äußere  $PT_1D$ -Lageregelkreis der unterlagerten Antriebsregelung*

$$\begin{cases} \dot{\mathbf{x}}_2 = -\mathbf{K}_{T_{l,2}}^{-1} \mathbf{x}_2 + \mathbf{K}_{T_{l,2}}^{-1} \mathbf{e}_2 \\ \mathbf{u}_2 = \mathbf{K}_{V_2} \left( \mathbf{I} - \mathbf{K}_{T_{r,2}} \mathbf{K}_{T_{l,2}}^{-1} \right) \mathbf{x}_2 + \mathbf{K}_{V_2} \mathbf{K}_{T_{r,2}} \mathbf{K}_{T_{l,2}}^{-1} \mathbf{e}_2 \end{cases} \quad (3.60)$$

mit  $\mathbf{e}_2 = \mathbf{q}_{\text{fwd}} - \mathbf{q}$  stabilisiert den inneren  $PI$ -Geschwindigkeitsregelkreis, definiert in Korollar 1, wenn

a) die  $PI$ -Geschwindigkeitsregelung exponentiell stabil mit der Ersatzzeitkonstante  $T_{e,1}$  ist

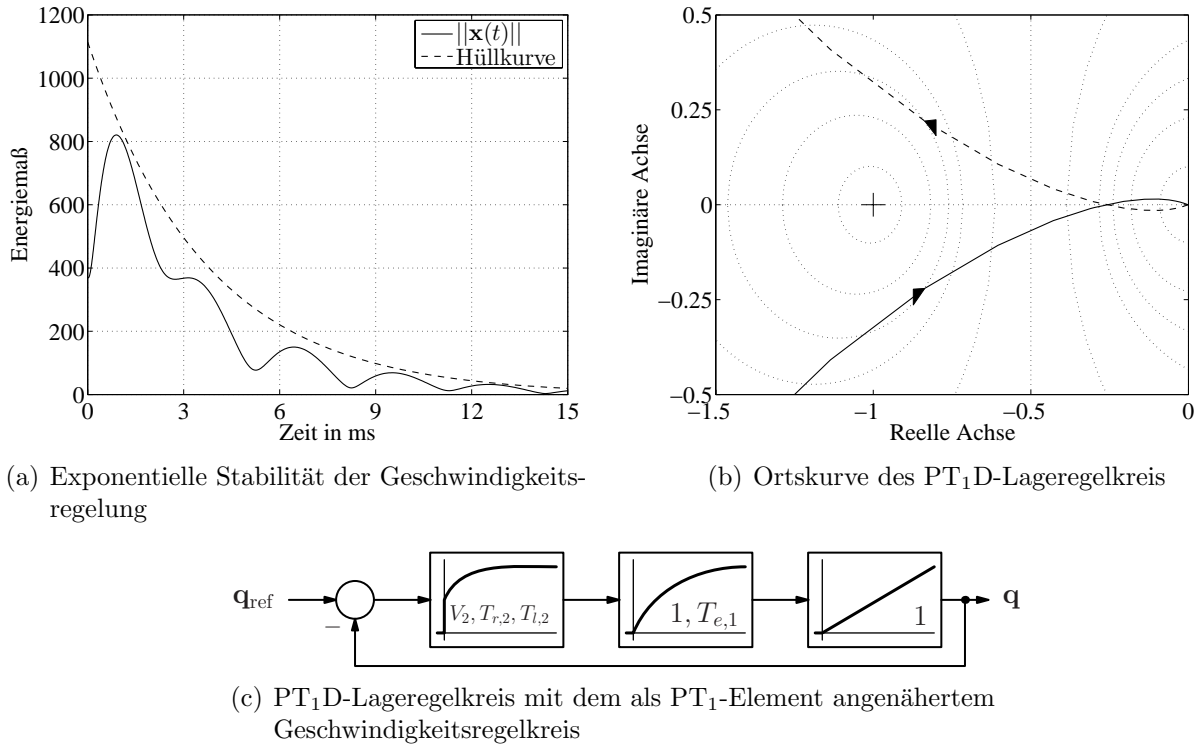


Abb. 3.7: Äußerer Lageregelkreis der unterlagerten Antriebsregelung

und

b) der Regelkreis, bestehend aus  $PT_1D$ -Lageregler, durch die Ersatzzeitkonstante  $T_{e,1}$  angenäherten  $PI$ -Geschwindigkeitsregler und Integrator nach den Methoden der linearen Regelungstechnik stabil ist.

**Beispiel 2 (Stabilitätsbeweis der  $PT_1D$ -Lageregelung)** Das Beispiel des Stabilitätsbeweises für die Lageregelung basiert auf den Annahmen im Beispiel 1. Die Auslegung des Lagereglers erfolgt nach den Gleichungen (3.11). Demnach ist die Verstärkung  $V_2$  und die Zeitkonstante  $T_{r,2}$  unabhängig von den Parametern des Geschwindigkeitsreglers. Einzig der Vorhalt  $T_{l,2}$  kompensiert die Integrationszeitkonstante.

Mit den definierten Parametern des Parallelroboters HEXA ist die Verstärkung  $V_2 = 138,89$ , die Zeitkonstante  $T_{r,2} = 2,4 \text{ ms}$  und der Vorhalt  $T_{l,2} = T_{i,1} = 4,8 \text{ ms}$ .

In Simulationen wurde ein Abklingkoeffizient der exponentiell stabilen  $PI$ -Geschwindigkeitsregelung von  $\lambda = 270$  und ein Faktor  $\alpha = 3$  ermittelt. Die Ersatzzeitkonstante des angenäherten Geschwindigkeitsregelkreises beträgt damit  $T_{e,1} = 1/\lambda = 37,04 \text{ ms}$ .

Im Rahmen dieses Beispiels wurde das Nyquistkriterium zur Überprüfung der Stabilität des  $PT_1D$ -Lageregelkreises angewandt. In Abbildung 3.7(b) ist die Ortskurve des Systems mit den oben bestimmten Parametern in dem kritischen Bereich dargestellt. Der Verlauf der Ortskurve umschließt den kritischen Punkt bei  $-1$  nicht, so dass die Stabilität nach dem

vereinfachten Nyquistkriterium gegeben ist.

Im Gegensatz zu Regelungskonzepten auf der Basis der exakten Linearisierung wird bei der hier angewandten Methode der berechneten Momente das Modell des Parallelroboters durch eine geeignete Vorsteuerung berücksichtigt. Im Sinne des Stabilitätsbeweises können diese im ungünstigsten Fall als Störgrößen aufgefasst werden. Sie haben allerdings keinen Einfluss auf die Struktur des geschlossenen Regelkreises und können deshalb die Stabilität der unterlagerten Antriebsregelung nicht beeinflussen.

### 3.4 Verhalten der Antriebsregelung

Die Führungssprungantwort der unterlagerten Antriebsregelung bestimmt die Dynamik des gesamten Systems und damit die Leistungsfähigkeit der Steuerungsarchitektur und schließlich die der Kontaktregelung. In diesem Abschnitt werden die grundlegenden Beziehungen zur Beschreibung der Dynamik der unterlagerten Antriebsregelung entwickelt, die schließlich in den Algorithmen zur Auslegung der Kontaktregelung ihre Anwendung finden.

Die Führungssprungantwort wird von den Verzögerungen der Umrichter  $T_c$  und der Geschwindigkeitsmessung  $T_m$  bestimmt. Fehler in der Adaption des Massenträgheitsmoments gemäß Gleichung (3.13) beeinflussen die Führungssprungantwort zusätzlich. Die Abweichungen des angenäherten Trägheitsmoments vom tatsächlich wirksamen Moment des  $k$ -ten Antriebs wird durch die Einführung eines multiplikativen Unsicherheitsfaktors  $\Delta^{(k)}$  beschrieben, so dass die Beziehung

$$\hat{j}_1^{(k)}(\mathbf{q}) = \Delta^{(k)} \hat{j}_1^{(k)}(\mathbf{q}) \quad \text{mit} \quad k \in [1; n]_{\mathbb{N}} \quad \text{und} \quad \Delta^{(k)} \in \left[ \Delta_{\min}^{(k)}; \Delta_{\max}^{(k)} \right]_{\mathbb{R}} \quad (3.61)$$

gilt. Die Führungsübertragungsfunktion der unterlagerten Antriebsregelung in den Gleichungen (3.11) mit Verzögerung der Vorsteuergrößen nach Gleichung (3.19) und (3.20) führt auf ein System sechster Ordnung:

$$F_{gw}^{(k)}(s) = \frac{(T_m s + 1)(4T_c s + 1)}{(T_c s + 1) \left[ 256\Delta^{(k)} T_m T_c^4 s^5 + 256\Delta^{(k)} T_c^3 (1 + T_m T_c) s^4 \right.} \dots \quad (3.62)$$

$$\left. \dots \frac{\dots}{+ 256\Delta^{(k)} T_c^3 s^3 + 4T_c (24T_c + T_m) s^2 + (16T_c + T_m) s + 1} \right]$$

Die nominelle Übertragungsfunktion  $F_{gw,n}$  ergibt sich aus obiger Gleichung (3.62), wenn das angenäherte Massenträgheitsmoment mit dem tatsächlichen Massenträgheitsmoment übereinstimmt, d. h.  $\Delta^{(k)} = 1$  ist.

Die, gemäß dem Stabilitätsbeweis in Abschnitt 3.3, asymptotisch stabile unterlagerte Antriebsregelung kapselt den Parallelroboter und garantiert eine durch die Übertragungsfunk-

tion (3.62) beschriebene Dynamik. Sie erfüllt damit alle Voraussetzungen für einen Einsatz in der Steuerungsarchitektur aus Kapitel 2.



## Kapitel 4

# Systemdynamik im Umgebungskontakt

**B**ERÜHRT der Endeffektor eines Parallelroboters oder ein fest damit verbundenes Werkstück im Arbeitsraum die Umgebung, dann treten externe Kräfte und Momente auf, die das Verhalten des Systems beeinflussen. Das Problem der Bestimmung der Richtung und Größe dieser Kräfte bzw. Momente sowie deren Wirkung auf die Bewegungen des Parallelroboters wird als Systemdynamik im Umgebungskontakt bezeichnet. Die Lösung dieser Fragestellung ist eine notwendige Voraussetzung sowohl für die Synthese der Kontaktregelung als auch für die Planung einer hybriden Regelungsaufgabe.

Die Richtung der externen, holonomen Kräfte im Kontaktpunkt wird durch die Geometrie der Umgebungsgrenze vorgegeben. Dieser Zusammenhang wird bereits in der wegweisenden Publikation von McClamroch und Wang [1988] beschrieben. Die Autoren entwickeln darin eine Beschreibung der Dynamik, in der, bei exakter Kenntnis der Umgebungsgeometrie, die beschränkten und unbeschränkten Freiheitsgrade des Systems zu jedem Zeitpunkt voneinander entkoppelt sind. Der gleiche Ansatz wird in [Mills und Goldenberg, 1989] angewendet, und führt hier auf ein Dynamikmodell in Form eines Beschreibungsvariablensystems. Den oben genannten Arbeiten ist gemein, dass sie eine exakte Beschreibung der Umgebungsgeometrie benötigen. Dies widerspricht der Intention einer Robotersteuerung auf der Basis von Aktionsprimitiven, die statt dessen sensorgeführte Bewegungen in einer unbekannten Umgebung realisiert.

Arbeiten von Seul Jung u. a. [2004] sowie Wang u. a. [2004] berücksichtigen dagegen Unsicherheiten im Umgebungsmodell, geben aber eine Struktur des Regelkreises vor, der für die vorgestellte Implementierung der Aktionsprimitive nicht angewendet werden kann, da die Konzepte Elemente der exakten Linearisierung enthalten.

Eine umfassende Betrachtung der Kontaktregelung teilt diese in die Phasen der Freiraumbewegung, der Transition und der eigentlichen Kraft- bzw. Momentenregelung ein [vgl. Brogliato, 1999]. Den Phasen der Transition und Kraft-Momentenregelung werden in dieser Arbeit unterschiedliche Modelle des Umgebungskontakts zugeordnet. Das Transitionsmodell beschreibt die impulsförmig auftretenden Reaktionen, wenn der Endeffektor des Parallelroboters auf die Umgebung aufprallt. Demgegenüber integriert das Kontaktmodell die statischen und dynamischen Eigenschaften der Umgebung nachdem der Endeffektor sicher auf der Umgebung ruht. Zusammengefasst bilden beide das Umgebungsmodell.

In den folgenden Abschnitten wird eine für das Aktionsprimitivkonzept geeignete Be-

schreibung der Richtung der externen Kräfte und Momente durch die Umgebungsgeometrie entwickelt. Zusammen mit dem Umgebungsmodell führt dies schließlich zur Systemdynamik im Umgebungskontakt.

## 4.1 Umgebungsgeometrie

Das Vorhandensein von Werkstücken und Zuführeinrichtungen im Arbeitsraum des Parallelroboters ist eine zur Ausführung einer Montageaufgabe notwendige Voraussetzung. In ihrer Gesamtheit werden die Objekte im Arbeitsraum als *Umgebung* bezeichnet, die den Arbeitsraum einschränken und Zwangsbedingungen etablieren. Die Umgebungsgeometrie bestimmt die Richtung der auftretenden externen Kräfte bzw. Momente. Sie wird durch eine endliche Anzahl glatter, eindimensionaler Teilumgebungen beschrieben [siehe Brogliato, 1999, Def. 1.6, S. 19]:

$$\Gamma_i(\mathbf{r}) = f_1(x_1) + f_2(x_2) + f_3(x_3) + x_0 = 0 \quad \text{mit} \quad i = [1; m]_{\mathbb{N}} \quad (4.1)$$

Dabei ist  $\mathbf{r} \in \mathbb{R}^{n_t}$  der Ortsvektor im Roboterkoordinatensystem BF und  $n_t$  die Anzahl der translatorischen Freiheitsgrade. Die Elemente  $x_{n_t}$  sind mit der glatten Funktion  $f_{n_t}$  gewichtet und ein konstanter Faktor  $x_0$  modifiziert die Geometrie zusätzlich. Die Zahl  $m$  symbolisiert die Anzahl der glatten Teilumgebungen. Gleichung

$$\Gamma_i(\mathbf{r}) \geq 0 \quad (4.2)$$

beschreibt somit die Umgebungsgrenzen und definiert den Unterraum  $\Phi$  des Arbeitsraums mit

$$\Phi = \bigcap_{i=1}^m \{\mathbf{r} : \Gamma_i(\mathbf{r}) \geq 0\}. \quad (4.3)$$

In diesem kann der Parallelroboter nur beschränkte Bewegungen ausführen. Die Umgebungsgrenze oder Oberfläche wird durch  $\partial\Phi$  symbolisiert. Auf die Umgebung  $\Phi$  sollen, von Seiten des Endeffektors, externe Kräfte und Momente ausgeübt werden. Im Rahmen dieser Arbeit wird die Kontaktregelung für den reibungslosen unilateralen Umgebungskontakt  $m = 1$  betrachtet. Demzufolge berührt der Endeffektor des Parallelroboters zu jedem Zeitpunkt der Kontaktregelung höchstens eine Teilumgebung.

Die Aufgabenplanung zur hybriden Lage- und Kontaktregelung findet zweckmäßiger Weise im kartesischen Roboterkoordinatensystem BF des Parallelroboters statt, da hier die Abmessungen und Umgebungsgrenzen intuitiv benannt werden können. Sollte aus beliebigen Gründen ein anderes Koordinatensystem verwendet werden, so sind die entsprechenden Positionen und Orientierungen in das Roboterkoordinatensystem zu transformieren. Der Referenzpunkt der Aufgabenplanung ist der Kontaktpunkt  $\mathbf{r}_c$ , an dem der Endeffektor des Parallelroboters

oder ein in irgendeiner Weise fest mit diesem verbundenes Werkstück für die gesamte Dauer des Aktionsprimitivs die Umgebung berühren soll. Die Bahn<sup>1</sup> des Kontaktpunkts und die externen Kräfte werden maßgeblich von der vorhandenen Umgebungsgeometrie beeinflusst. Für einen unilateralen Umgebungskontakt können im Roboterkoordinatensystem nur Bahnen des Ortsvektors  $\mathbf{r}_c$  im  $(n_t - 1)$ -dimensionalen Raum vorgegeben werden. Die verbleibende abhängige Koordinate wird von der Umgebungsgeometrie bestimmt.

Die geplante Bahnkurve des Kontaktpunkts  $\mathbf{r}_{\text{des},c}(t)$  mit  $t$  als Parameter in  $n_t - 1$  Freiheitsgraden lautet

$$\mathbf{r}_{\text{des},c}(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_{k-1}(t) \\ x_{k+1}(t) \\ \vdots \\ x_{n_t}(t) \end{bmatrix} \quad \text{mit} \quad \mathbf{r}_{\text{des},c} \in \mathbb{R}^{(n_t-1)} \quad \text{und} \quad 1 \leq k \leq n_t. \quad (4.4)$$

Dabei ist  $k$  der Index des abhängigen Freiheitsgrads. Unter Vernachlässigung der Deformation modifiziert die Umgebungsgeometrie die Bahn nach folgender Gesetzmäßigkeit

$$\mathbf{r}_c(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_{k-1}(t) \\ f_k^{-1} \left( - \sum_{\substack{i=1 \\ i \neq k}}^{n_t} f_i(x_i) - x_0 \right) \\ x_{k+1}(t) \\ \vdots \\ x_{n_t}(t) \end{bmatrix} \quad (4.5)$$

mit  $\mathbf{r}_c \in \mathbb{R}^{n_t}$ .

Im Umgebungskontakt wirken externe Kräfte auf den Kontaktpunkt  $\mathbf{r}_c$ . Die Endeffektorkräfte zeigen in Richtung des nach außen gerichteten Normalenvektors

$$\mathbf{n}_c(t) = \frac{\nabla \Gamma(\mathbf{r}_c(t))}{\|\nabla \Gamma(\mathbf{r}_c(t))\|} \quad (4.6a)$$

der senkrecht zur Grenze der Teilumgebung  $\partial\Phi$  steht. Der Tangentialvektor

$$\tilde{\mathbf{t}}_c(t) = \frac{d\mathbf{r}_c(t)}{dt} \quad (4.6b)$$

---

<sup>1</sup>Die Bahn ist im mathematischen Sinn eine Raumkurve, durch die alle Orte beschrieben werden, die der Kontaktpunkt passiert. Demgegenüber ist bei einer Trajektorie jedem Ort ein eindeutiger Zeitpunkt zugeordnet, so dass die Geschwindigkeit und Beschleunigung ebenfalls festgelegt sind.

zeigt in die Bewegungsrichtung des Endeffektors. Normiert lautet dieser  $\mathbf{t}_c(t) = \frac{\tilde{\mathbf{t}}_c(t)}{\|\tilde{\mathbf{t}}_c(t)\|}$  und definiert gemeinsam mit dem Binormalvektor

$$\mathbf{b}_c(t) = \mathbf{n}_c(t) \times \mathbf{t}_c(t) \quad (4.6c)$$

eine Ebene, in der Reibungskräfte auf den Endeffektor wirken können. Die drei Vektoren (4.6a) bis (4.6c) spannen ein rechtshändiges Koordinatensystem auf. Zusammen mit dem Kontaktpunkt im Roboterkoordinatensystem ergibt sich für jeden Zeitpunkt  $t$  eine homogene Transformationsmatrix,

$${}^{\text{BF}}\mathbf{T}_{\text{IF}}(t) = \begin{bmatrix} \mathbf{t}_c(t) & \mathbf{b}_c(t) & \mathbf{n}_c(t) & {}^{\text{BF}}\mathbf{r}_c(t) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_c(t) & {}^{\text{BF}}\mathbf{r}_c(t) \\ \mathbf{0}^T & 1 \end{bmatrix}, \quad (4.7)$$

die das Kontaktkoordinatensystem IF<sup>2</sup> definiert.

Befindet sich der Endeffektor oder ein Werkstück während einer hybriden Bewegung ständig im Umgebungskontakt, führt der Parallelroboter eine *Kontaktbewegung* aus.

In Abbildung 4.1 ist die Verformung der Endeffektorbahn durch einen Umgebungskontakt dargestellt. Der Parallelroboter befindet sich in der Kugel. Die geplante Bahn  $\mathbf{r}_{\text{des},c}(t)$  verläuft in der Projektion der kugelförmigen Umgebung auf die  $x$ - $y$ -Ebene von Punkt  $t_a$  nach  $t_b$  und wird im Verlauf der hybriden Lage- und Kontaktregelung von der Umgebung verformt. Der tatsächliche Verlauf von  $t'_a$  auf der Innenseite der Kugel nach  $t'_b$  wird durch den Tangentialvektor  $\mathbf{t}_c(t')$  und den Binormalvektor  $\mathbf{b}_c(t')$  beschrieben. Die vorhandenen externen Kontakt- und Reibungskräfte addieren sich zu einer resultierenden Kraft, die auf dem angedeuteten Reibkegel liegt.

## 4.2 Kräfte und Momente

Die Richtung der externen Kräfte ist durch die geplante Bahn und die Umgebungsgeometrie festgelegt. Ein unilateraler und reibungsfreier Umgebungskontakt verursacht nur eine Kraftkomponente parallel zum Normalenvektor  $\mathbf{n}_c$ . Im Kontaktkoordinatensystem ist dies gleichbedeutend mit der  $z$ -Achse. Die Richtung der Kontaktkräfte kann daher wie folgt angenommen werden:

**Annahme 1 (Kraft im Kontaktkoordinatensystem)** Der *generalisierte Kraft-Momentenvektor* im Kontaktkoordinatensystem IF lautet

$${}^{\text{IF}}\mathbf{f} = \begin{bmatrix} \zeta \\ \boldsymbol{\mu} \end{bmatrix} = {}^{\text{IF}} \begin{bmatrix} 0 & 0 & f_z & 0 & 0 & 0 \end{bmatrix}^T. \quad (4.8)$$

<sup>2</sup>Die naheliegende Abkürzung CF für *contact frame* wurde hier vermieden, um eine Verwechslung mit dem in diesem Thema immer präsenten *compliance frame* nach [Bruyninckx und De Schutter, 1996] zu vermeiden. Statt dessen wurde auf die Abkürzung IF für *interaction frame* zurückgegriffen.

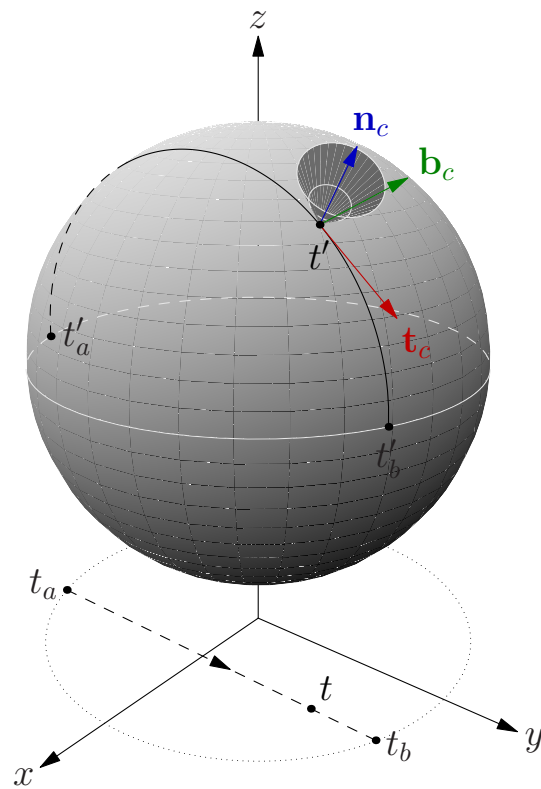


Abb. 4.1: Verformung der Bahn des Kontaktpunkts  $\mathbf{r}_c(t)$  durch eine kugelförmige Umgebung. Die Achsen des Kontaktkoordinatensystems zeigen in Richtung der Endeffektorkräfte.

Dabei sind  ${}^{\text{IF}}\boldsymbol{\zeta}$  die Kräfte parallel zu den Koordinatenachsen und  ${}^{\text{IF}}\boldsymbol{\mu}$  die Momente im Kontaktkoordinatensystem.

Da der Kraftvektor  ${}^{\text{IF}}\boldsymbol{\zeta}$  nur ein Element besitzt, wird die abkürzende Schreibweise

$${}^{\text{IF}}\boldsymbol{\zeta} = \mathbf{s}^{\text{IF}} f_z \quad \text{mit} \quad \mathbf{s} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$$

eingeführt.

### 4.2.1 Kräfte und Momente im Roboterkoordinatensystem

Die unterlagerte Antriebsregelung des Parallelroboters findet im Roboterkoordinatensystem BF statt. Mit Kenntnis der im Abschnitt 4.1 hergeleiteten Beschreibung des Kontaktkoordinatensystems IF können die Kräfte und Momente im Roboterkoordinatensystem notiert werden

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \\ {}^{\text{BF}}\mathbf{r}_c \times {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \end{bmatrix}. \quad (4.9)$$

Die Einführung des Matrixoperators  $\mathbf{S}(\mathbf{r}) = \begin{bmatrix} 0 & -x & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$  nach Sciavicco und Siciliano [2000] erlaubt die abkürzende Schreibweise

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \\ \mathbf{S}({}^{\text{BF}}\mathbf{r}_c) {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \end{bmatrix} \quad (4.10)$$

und schließlich die Definition der Jacobimatrix  ${}^{\text{BF}}\mathbf{J}_{\text{IF}} \in \mathbb{R}^{6 \times 3}$ , so dass

$${}^{\text{BF}}\mathbf{f} = {}^{\text{BF}}\mathbf{J}_{\text{IF}}({}^{\text{BF}}\mathbf{x}_c) {}^{\text{IF}}\boldsymbol{\zeta} \quad \text{mit} \quad {}^{\text{BF}}\mathbf{J}_{\text{IF}}({}^{\text{BF}}\mathbf{x}_c) := \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{IF}} \\ \mathbf{S}({}^{\text{BF}}\mathbf{r}_c) {}^{\text{BF}}\mathbf{R}_{\text{IF}} \end{bmatrix} \quad (4.11)$$

gilt. In obiger Gleichung ist  ${}^{\text{BF}}\mathbf{r}_c$  der zuvor definierte Kontaktpunkt im Roboterkoordinatensystem BF,  ${}^{\text{BF}}\mathbf{R}_{\text{IF}}$  die zugehörige Rotationsmatrix und  ${}^{\text{IF}}\boldsymbol{\zeta}$  die in der Gleichung (4.8) auftretenden Kräfte.

### 4.2.2 Kräfte und Momente im Task-Frame

Die hybride Lage- und Kontaktregelung des Parallelroboters findet im Task-Frame TF statt. Die dort auftretenden Kräfte und Momente werden somit von der Transformation des Kontaktkoordinatensystems IF in den Task-Frame TF bestimmt und beeinflussen die Aufgabenplanung. Gemäß den Definitionen der in Abschnitt 2.1 beschriebenen Roboterprogram-

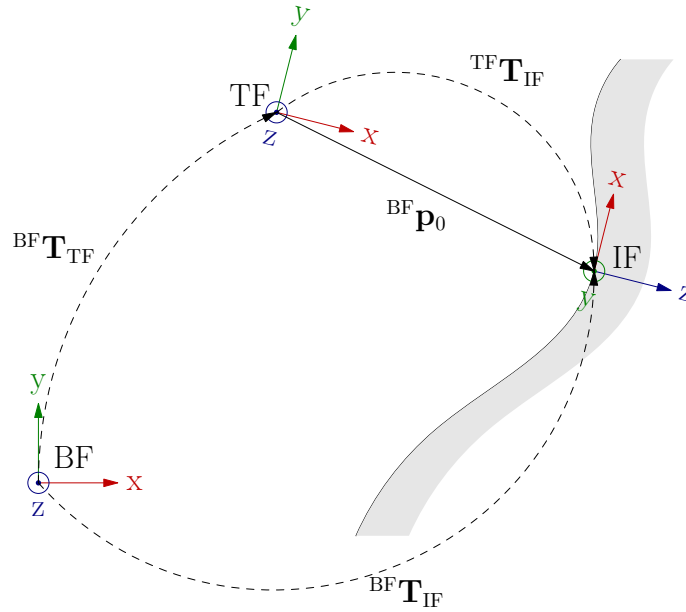


Abb. 4.2: Darstellung der relevanten Koordinatensysteme einer hybriden Regelung mit Umgebungskontakt

mierung existieren verschiedene Möglichkeiten den Verlauf des Task-Frames innerhalb eines Aktionsprimitivs zu bestimmen. Beinhaltet die hybride Regelungsaufgabe eine Kontaktregelung, ist die Unterscheidung der folgenden zwei Fälle relevant:

Fall 1: Der Task-Frame wird willkürlich festgelegt. Als Sonderfälle sind hier die Definition eines ortsfesten Task-Frames sowie die Verankerung des Task-Frames mit einem beliebigen externen Koordinatensystem enthalten.

Fall 2: Der Task-Frame wird mit einem roboterinternen Koordinatensystem verankert und folgt so z. B. mit konstantem Abstand einer Umgebungsgrenze.

Abbildung 4.2 veranschaulicht die Beziehungen der Koordinatensysteme eines Parallelroboters im Umgebungskontakt. Im Fall des willkürlich definierten Task-Frames (Fall 1) ist die Transformation  ${}^{\text{BF}}\mathbf{T}_{\text{TF}}(t)$  bekannt. Durch geeignete Wahl der Reglertypen und Regelziele im Aktionsprimitiv ist sichergestellt, dass der Kontaktpunkt die Umgebungsgrenze nicht verlässt und sich auf einer gemäß Gleichung (4.7) definierten Bahn bewegt. Die im Task-Frame wirkenden Kräfte und Momente werden durch die Transformation

$${}^{\text{TF}}\mathbf{T}_{\text{IF}}(t) = {}^{\text{BF}}\mathbf{T}_{\text{TF}}^{-1}(t) {}^{\text{BF}}\mathbf{T}_{\text{IF}}(t) = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} & {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T \left( {}^{\text{BF}}\mathbf{r}_c - {}^{\text{BF}}\mathbf{r}_{\text{TF}} \right) \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (4.12)$$

bestimmt und lauten

$$\begin{aligned} {}^{\text{TF}}\mathbf{f} &= \begin{bmatrix} {}^{\text{TF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \\ {}^{\text{TF}}\mathbf{r}_{\text{IF}} \times \left( {}^{\text{TF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \right) \end{bmatrix} \\ &= \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \\ \left( {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T \left( {}^{\text{BF}}\mathbf{r}_c - {}^{\text{BF}}\mathbf{r}_{\text{TF}} \right) \right) \times \left( {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} \right) \end{bmatrix}. \end{aligned} \quad (4.13)$$

In den obigen Gleichungen stellt  ${}^{\text{BF}}\mathbf{R}_{\text{TF}}$  die Rotationsmatrix des Task-Frames im Roboterkoordinatensystem dar.  ${}^{\text{BF}}\mathbf{r}_{\text{TF}}$  ist der zugehörige Ortsvektor. Die Transformation des Kontaktkoordinatensystems wurde in Gleichung (4.7) definiert.

Ist der Task-Frame des Aktionsprimitivs mit einem roboterinternen Koordinatensystem verankert (Fall 2), kann dessen Position  ${}^{\text{BF}}\mathbf{r}_{\text{TF}}(t_0)$  nur zu Beginn der Regelungsaufgabe festgelegt werden. Wie in Abbildung 4.2 zu sehen, ist der Abstand von Task-Frame und Kontaktkoordinatensystem zum Zeitpunkt  $t_0$

$${}^{\text{BF}}\mathbf{p}_0 = {}^{\text{BF}}\mathbf{r}_c(t_0) - {}^{\text{BF}}\mathbf{r}_{\text{TF}}(t_0). \quad (4.14)$$

Da sowohl der Task-Frame als auch der Kontaktpunkt  ${}^{\text{BF}}\mathbf{r}_c(t)$  mit dem Roboter verankert sind, bewegt sich der Ortsvektor des Task-Frames im Verlauf der Regelung in Abhängigkeit von der Umgebungsgeometrie entlang der Bahn

$${}^{\text{BF}}\mathbf{r}_{\text{TF}}(t) = {}^{\text{BF}}\mathbf{r}_c(t) - {}^{\text{BF}}\mathbf{p}_0. \quad (4.15)$$

Die Orientierung des Task-Frames  ${}^{\text{BF}}\mathbf{R}_{\text{TF}}(t)$  wird für den gesamten Verlauf der Regelungsaufgabe definiert. Sie kann als konstant oder zeitabhängig angenommen werden. Die Verknüpfungen mit dem Hand- oder Kontaktkoordinatensystem sind darin enthaltene Sonderfälle. Die Bahn des Task-Frames ist damit

$${}^{\text{BF}}\mathbf{T}_{\text{TF}}(t) = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}}(t) & {}^{\text{BF}}\mathbf{r}_c(t) - {}^{\text{BF}}\mathbf{p}_0 \\ \mathbf{0}^T & 1 \end{bmatrix}. \quad (4.16)$$

Die im Task-Frame wirkenden Kräfte und Momente werden durch die Transformation

$${}^{\text{TF}}\mathbf{T}_{\text{IF}}(t) = {}^{\text{BF}}\mathbf{T}_{\text{TF}}^{-1}(t) {}^{\text{BF}}\mathbf{T}_{\text{IF}}(t) = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} & {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{p}_0 \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (4.17)$$



bestimmt und lauten

$${}^{\text{TF}}\mathbf{f} = \begin{bmatrix} {}^{\text{TF}}\mathbf{R}_{\text{IF}}^T {}^{\text{IF}}\boldsymbol{\zeta} \\ {}^{\text{TF}}\mathbf{r}_{\text{IF}} \times {}^{\text{IF}}\boldsymbol{\zeta} \end{bmatrix} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}}^T {}^{\text{IF}}\boldsymbol{\zeta} \\ \left( {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{p}_0 \right) \times \left( {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}}^T {}^{\text{IF}}\boldsymbol{\zeta} \right) \end{bmatrix}. \quad (4.18)$$

Für die Gleichungen (4.13) und (4.18) gelten die in Beziehung (4.8) definierten Bedingungen, so dass zu jedem Zeitpunkt der Kontaktregelung von einem unilateralen Umgebungskontakt ausgegangen werden kann.

## 4.3 Umgebungsmodell

Die externen Kräfte und Momente, die auf den Endeffektor des Parallelroboters wirken sind vektorielle Größen. Während die Umgebungsgeometrie ihre Richtung festlegt, wird deren Wert in der Transitionsphase vom Transitionsmodell und im Verlauf des Umgebungskontakts vom Kontaktmodell bestimmt. Beide Modelle werden im Umgebungsmodell zusammengefasst.

### 4.3.1 Kontaktmodell

Das Kontaktmodell beschreibt den Einfluss der Lageänderungen im Roboterkoordinatensystem BF auf die Kräfte und Momente im Kontaktkoordinatensystem IF, wenn alle transienten Vorgänge abgeschlossen sind und der Endeffektor sicher im Umgebungskontakt verharret. Die Komplexität der Modelle reicht von statischen Modellen, die eine einfache Umgebungssteifigkeit annehmen [Sciavicco und Siciliano, 2000], bis hin zu dynamischen Modellen der Kontaktsituation [Mills und Lokhorst, 1993]. Im Rahmen dieser Arbeit wird ein modular aufgebautes Kontaktmodell verwendet, das, beginnend mit der statischen Modellierung, zu einem komplexen dynamischen Verhalten erweitert werden kann.

#### Statisches Kontaktmodell

Ausgangspunkt der Überlegungen und Motivation des statischen Kontaktmodells ist die Federkraft. Das statische Kontaktmodell bildet die Endeffektorkräfte proportional zur Verschiebung der Umgebungsgrenze ab. Der Proportionalitätsfaktor ist die Umgebungssteifigkeit  $c_{e,0}$  und wird in der Einheit  $[c_{e,0}] = \text{N/m}$  angegeben. Die Umgebung besteht aus starren Körpern, deren Oberfläche nicht deformiert werden kann. Übt der Endeffektor des Parallelroboters eine Kraft auf die Umgebung aus, wird die Umgebungsgrenze verschoben. Die Orientierung ändert sich bei diesem Vorgang nicht, so dass die Richtung der Kraft konstant bleibt.

In einem Freiheitsgrad ist die Endeffektorkraft bei einer translatorischen Kontaktbewe-

gung

$$f = \begin{cases} 0 : & \Gamma(x) < 0, \\ c_{e,0}(x - x_0) : & \Gamma(x) \geq 0. \end{cases} \quad (4.19)$$

Die Umgebungsgeometrie wird durch  $\Gamma(x) = x - x_0$  beschrieben, wobei  $x_0$  die Umgebungsgrenze darstellt. Befindet sich der Endeffektor im Freiraum, wird keine Kraft ausgeübt. Im Umgebungskontakt wirkt eine Kraft, die durch die Verschiebung der Oberfläche  $x - x_0$  und die Umgebungssteifigkeit  $c_{e,0}$  bestimmt wird.

Das Moment einer rotatorischen Kontaktbewegung ist, neben der Verschiebung der Oberfläche, von der Länge des wirksamen Hebelarms  $l$  abhängig. Eine Drehung um den Winkel  $\varphi$  verschiebt den Kontaktpunkt um

$$\Delta z = l \sin \varphi$$

parallel zum Normalenvektor der Umgebung und verursacht eine Kontaktkraft. Das Moment, das aufgebracht werden muss, um dieser Kraft entgegen zu wirken, ist

$$\mu = \begin{cases} 0 : & \Gamma(\varphi) < 0, \\ c_{e,0}l^2 \sin(\varphi - \varphi_0) : & \Gamma(\varphi) \geq 0, \end{cases} \quad (4.20)$$

In obiger Gleichung definiert  $\Gamma(\varphi) = \varphi - \varphi_0$  die Umgebungsgeometrie. Für kleine Winkel gilt die Annahme  $\sin \varphi \approx \varphi$ , so dass (4.20) zu

$$\mu = \begin{cases} 0 : & \Gamma(\varphi) < 0 \\ c_{e,0}l^2(\varphi - \varphi_0) : & \Gamma(\varphi) \geq 0 \end{cases} \quad (4.21)$$

vereinfacht werden kann. Die ebenfalls vorhandene Verschiebung des Kontaktpunkts von

$$\Delta y = l(1 - \cos \varphi)$$

tangential zur Oberfläche kann für kleine Winkel  $\varphi$  vernachlässigt werden.

Die Verschiebung der Oberfläche im Task-Frame-Formalismus ist stark vergrößert in Abbildung 4.3 veranschaulicht. Am Beginn der Kraft-Momentenregelung stellt sich die Situation wie folgt dar: Die Lage des Endeffektors im Roboterkoordinatensystem ist durch die Transformation  ${}^{\text{BF}}\mathbf{T}_0$  gegeben. Eine starre Verbindung zwischen Endeffektor und Kontaktpunkt wird durch  ${}^0\mathbf{T}_c$  beschrieben. Reglereingriffe verändern die Lage des Endeffektors gemäß

$$d\mathbf{x} = \begin{bmatrix} d\mathbf{r}^T & d\boldsymbol{\varphi}^T \end{bmatrix}^T = {}^{\text{BF}}\mathbf{x}_1 - {}^{\text{BF}}\mathbf{x}_0.$$

In obiger Gleichung ist  ${}^{\text{BF}}\mathbf{x}_0$  die Lage des Endeffektors im Umgebungskontakt zu Beginn der

Regelung und  ${}^{\text{BF}}\mathbf{x}_1$  die Lage aufgrund des Reglereingriffs. Die entsprechende Transformation ist

$${}^0\mathbf{T}_1(d\mathbf{x}) = \begin{bmatrix} {}^0\mathbf{R}(d\boldsymbol{\varphi}) & {}^0d\mathbf{r} \\ \mathbf{0}^T & 1 \end{bmatrix}. \quad (4.22)$$

Die verschobene Lage der Oberfläche im Roboterkoordinatensystem lautet

$${}^{\text{BF}}\mathbf{T}_d = {}^{\text{BF}}\mathbf{T}_0 {}^0\mathbf{T}_1(d\mathbf{x}) {}^1\mathbf{T}_d \quad (4.23)$$

und kann in Abbildung 4.3 nachvollzogen werden. Dabei ist  ${}^1\mathbf{T}_d$  die Transformation, die die Verbindung zwischen der Endlage des Endeffektors und dem Kontaktpunkt definiert. Diese bleibt während der Regelung unverändert und ist identisch mit  ${}^0\mathbf{T}_c$ , so dass obige Gleichung zu

$${}^{\text{BF}}\mathbf{T}_d = {}^{\text{BF}}\mathbf{T}_0 {}^0\mathbf{T}_1(d\mathbf{x}) {}^0\mathbf{T}_c \quad (4.24)$$

wird. Nach der homogenen Transformation gilt explizit

$${}^{\text{BF}}\mathbf{T}_d = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_0 {}^0\mathbf{R}(d\boldsymbol{\varphi}) {}^0\mathbf{R}_c & {}^{\text{BF}}\mathbf{R}_0 {}^0\mathbf{R}(d\boldsymbol{\varphi}) {}^0\mathbf{r}_c + {}^{\text{BF}}\mathbf{R}_0 {}^0d\mathbf{r} + {}^{\text{BF}}\mathbf{r}_0 \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (4.25)$$

wobei

$${}^{\text{BF}}\mathbf{r}_d = \mathbf{D}^T {}^{\text{BF}}\mathbf{T}_d \mathbf{t} = {}^{\text{BF}}\mathbf{R}_0 {}^0\mathbf{R}(d\boldsymbol{\varphi}) {}^0\mathbf{r}_c + {}^{\text{BF}}\mathbf{R}_0 {}^0d\mathbf{r} + {}^{\text{BF}}\mathbf{r}_0 \quad (4.26)$$

den Ortsvektor des verschobenen Kontaktpunkts im Roboterkoordinatensystem darstellt. Die ebenfalls berechnete Orientierung  ${}^{\text{BF}}\mathbf{R}_d = \mathbf{D}^T {}^{\text{BF}}\mathbf{T}_d \mathbf{D}$  wird verworfen, da die Verschiebung der Oberfläche nur kleine Änderungen der Lage bewirkt. In dieser Epsilonumgebung wird die Oberfläche als Ebene angenommen. Die Orientierung und damit die Richtung der Kontaktkraft bleibt konstant, so dass

$${}^{\text{BF}}\mathbf{R}_d := {}^{\text{BF}}\mathbf{R}_c \quad (4.27)$$

und schließlich

$${}^{\text{BF}}\mathbf{T}_d := \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_c & {}^{\text{BF}}\mathbf{R}_0 {}^0\mathbf{R}(d\boldsymbol{\varphi}) {}^0\mathbf{r}_c + {}^{\text{BF}}\mathbf{R}_0 {}^0d\mathbf{r} + {}^{\text{BF}}\mathbf{r}_0 \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (4.28)$$

ist. Der sechsdimensionale Vektor mit allen Variablen der Lage des Kontaktpunkts im Ro-

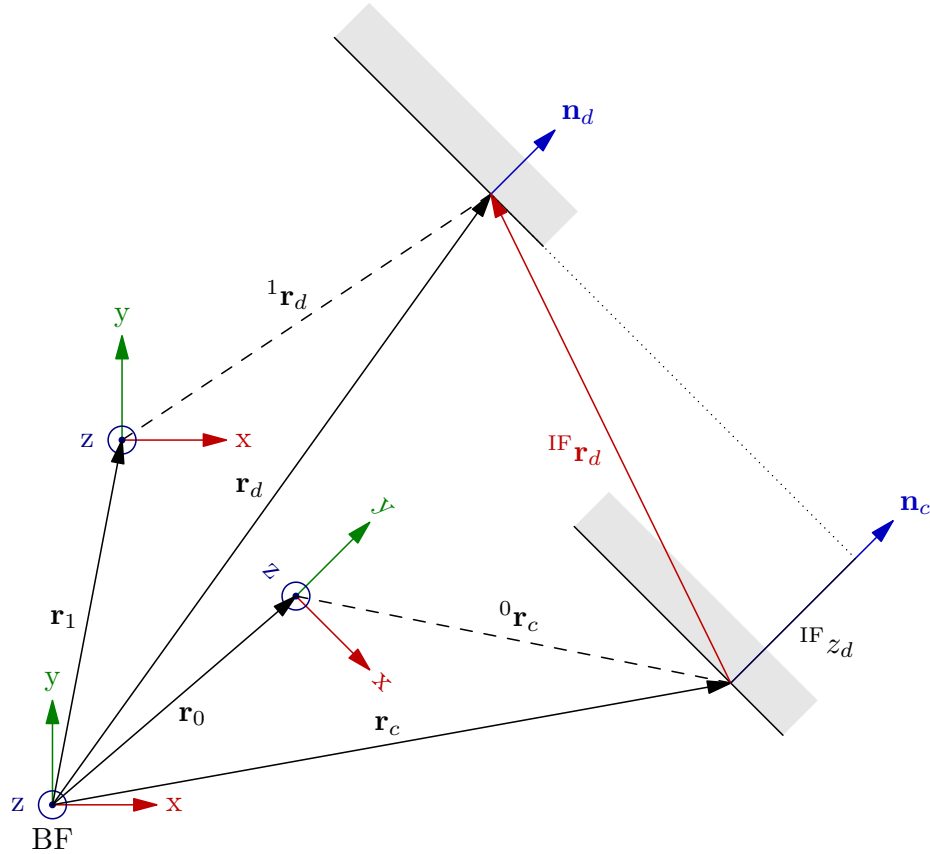


Abb. 4.3: Verschiebung der Umgebungsgrenze durch Bewegungen im Roboterkoordinatensystem BF (stark vergrößert)

boterkoordinatensystem lautet

$${}^{\text{BF}}\mathbf{x}_d = \begin{bmatrix} \mathbf{D}^T {}^{\text{BF}}\mathbf{T}_d({}^{\text{BF}}d\mathbf{x})\mathbf{t} \\ \text{RPY}(\mathbf{D}^T {}^{\text{BF}}\mathbf{T}_d({}^{\text{BF}}d\mathbf{x})\mathbf{D}) \end{bmatrix}. \quad (4.29)$$

Der Kontaktpunkt befindet sich im Umgebungskontakt, solange die Bedingung

$$\Gamma({}^{\text{BF}}\mathbf{r}_d) \geq 0 \quad (4.30)$$

erfüllt ist.

Die Verschiebung des Kontaktpunkts im Kontaktkoordinatensystem IF bestimmt die Auslenkung der Oberfläche parallel zum Normalenvektor  $\mathbf{n}_c$ . Über den Pfad vom ursprünglichen Kontaktpunkt über den Endeffektor zu Beginn der Regelung und weiter vom verschobenen Endeffektor zum verschobenen Kontaktpunkt lässt sich die Transformation  ${}^{\text{IF}}\mathbf{T}_d$  mit der Gleichung

$${}^{\text{IF}}\mathbf{T}_d = {}^0\mathbf{T}_c^{-1} {}^0\mathbf{T}_1(d\mathbf{x}) {}^0\mathbf{T}_c \quad (4.31)$$

bestimmen. Der Ortsvektor dieser Transformation ergibt sich zu

$${}^{\text{IF}}\mathbf{r}_d = \mathbf{D}^T {}^{\text{IF}}\mathbf{T}_d \mathbf{t} = {}^0\mathbf{R}_c^{-1} \left[ {}^0\mathbf{d}\mathbf{r} + \left( {}^0\mathbf{R}(\mathbf{d}\boldsymbol{\varphi}) - \mathbf{I} \right) {}^0\mathbf{r}_c \right] \quad (4.32)$$

und die Auslenkung entlang der  $z$ -Achse im Kontaktkoordinatensystem ist

$${}^{\text{IF}}z_d = \mathbf{s}^T \mathbf{D}^T {}^{\text{IF}}\mathbf{T}_d ({}^{\text{BF}}\mathbf{d}\mathbf{x}) \mathbf{t}. \quad (4.33)$$

Im verschobenen Kontaktpunkt  ${}^{\text{IF}}\mathbf{r}_d$  übt der Endeffektor des Parallelroboters eine Kraft von

$${}^{\text{IF}}\boldsymbol{\zeta} = \mathbf{s} {}^{\text{IF}}f_z = \mathbf{s} c_{e,0} {}^{\text{IF}}z_d \quad (4.34)$$

auf die Umgebung aus. Der wirksame Hebelarm  ${}^{\text{BF}}\mathbf{r}_d$  und die Rotation  ${}^{\text{BF}}\mathbf{R}_d$  transformieren diese Kraft schließlich in das Roboterkoordinatensystem BF

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_d {}^{\text{IF}}\boldsymbol{\zeta} \\ {}^{\text{BF}}\mathbf{r}_d \times ({}^{\text{BF}}\mathbf{R}_d {}^{\text{IF}}\boldsymbol{\zeta}) \end{bmatrix}. \quad (4.35)$$

Der Matrixoperator  $\mathbf{S}(\mathbf{r})$  erlaubt die abkürzende Schreibweise

$${}^{\text{BF}}\mathbf{f} = \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_d {}^{\text{IF}}\boldsymbol{\zeta} \\ \mathbf{S}({}^{\text{BF}}\mathbf{r}_d) {}^{\text{BF}}\mathbf{R}_d {}^{\text{IF}}\boldsymbol{\zeta} \end{bmatrix} \quad (4.36)$$

und schließlich die Definition der Jacobimatrix  ${}^{\text{BF}}\mathbf{J}_{\text{IF}} \in \mathbb{R}^{6 \times 3}$ , so dass

$${}^{\text{BF}}\mathbf{f} = {}^{\text{BF}}\mathbf{J}_{\text{IF}}({}^{\text{BF}}\mathbf{x}_d) {}^{\text{IF}}\boldsymbol{\zeta} \quad \text{mit} \quad {}^{\text{BF}}\mathbf{J}_{\text{IF}}({}^{\text{BF}}\mathbf{x}_d) := \begin{bmatrix} {}^{\text{BF}}\mathbf{R}_d \\ \mathbf{S}({}^{\text{BF}}\mathbf{r}_d) {}^{\text{BF}}\mathbf{R}_d \end{bmatrix} \quad (4.37)$$

gilt. Durch das oben beschriebene statische Kontaktmodell können die Kräfte und Momente  ${}^{\text{BF}}\mathbf{f}$ , die der Parallelroboter im Roboterkoordinatensystem aufbringen muss, um eine vorgegebene Lageänderung  ${}^{\text{BF}}\mathbf{d}\mathbf{x}$  auszuführen, bestimmt werden. Bestehende Ansätze in [Finckemeyer, 2004; Roy und Whitcomb, 2002] werden dabei um die explizite Berücksichtigung eines fest mit dem Task-Frame verbundenen Hebelarms erweitert. Dadurch ist die Berechnung der im Verlauf einer Kontaktregelung auftretenden Momente möglich und muss nicht in der Beschreibung der Umgebungsgeometrie implizit vorhanden sein.

### Dynamisches Kontaktmodell

Ein in Reihe zur einfachen Umgebungssteifigkeit geschaltetes Feder-Dämpfer-Masse-System erweitert das statische Kontaktmodell zu einem dynamischen Kontaktmodell. Durch diese Maßnahme wird eine zusätzliche Nachgiebigkeit in das Systemverhalten eingebracht. Das so entstandene Modell ist in Abbildung 4.4(a) mit diskreten Elementen dargestellt. Die Eingangsgröße ist die Verschiebung der Umgebungsgrenze in Richtung des Normalenvektors  ${}^{\text{IF}}z_d$ . Die Übertragungsfunktion im Kontaktkoordinatensystem IF lautet

$$G_E^{(z_d)}(s) = \frac{c_{e,0} (m_{e,1}s^2 + d_{e,1}s + c_{e,1})}{m_{e,1}s^2 + d_{e,1}s + c_{e,0} + c_{e,1}}. \quad (4.38)$$

Der Amplitudengang in Abbildung 4.4(b) verdeutlicht das Systemverhalten. Schnelle Änderungen der Endeffektorlage rufen Kontaktkräfte hervor, die hauptsächlich durch den statischen Teil des Modells mit der Federkonstanten  $c_{e,0}$  bestimmt werden. Die Nachgiebigkeit repräsentiert Elastizitäten in der Umgebung, die bei langsamen Lageänderungen und im stationären Fall überwiegen. Das Einschwingverhalten wird durch die Masse  $m_{e,1}$  und die Dämpfungskonstante  $d_{e,1}$  bestimmt. Die obere Resonanzfrequenz ist

$$\omega_1 = \sqrt{\frac{c_{e,0} + c_{e,1}}{m_{e,1}}}, \quad (4.39)$$

während die untere Eckfrequenz durch

$$\omega_{01} = \sqrt{\frac{c_{e,1}}{m_{e,1}}} \quad (4.40)$$

bestimmt wird. Im stationären Zustand vereinfacht sich das Kontaktmodell zu einer Reihenschaltung der Federn, so dass sich nach dem Endwertsatz<sup>3</sup> eine Umgebungssteifigkeit von

$$\lim_{s \rightarrow 0} G_E^{(z_d)}(s) = \frac{c_{e,0} \cdot c_{e,1}}{c_{e,0} + c_{e,1}} = c_{e,0} \parallel c_{e,1} \quad (4.41)$$

einstellt.

Die Berechnung der wirksamen Endeffektorkräfte und -momente im Roboterkoordinatensystem BF entspricht dem Vorgehen im Fall des statischen Kontaktmodells in Abschnitt 4.3.1. Mit Gleichung (4.32) wird die Verschiebung der Umgebungsgrenze  ${}^{\text{BF}}\mathbf{r}_d$  auf-

<sup>3</sup>Endwertsatz: Strebt die Funktion  $f(t)$  für  $t \rightarrow \infty$  einem endlichen Grenzwert zu, so gilt  $\lim_{t \rightarrow \infty} f(t) = \lim_{s \rightarrow 0} sG(s)U(s)$  [vgl. Föllinger, 1990, Kap. 17.1.4]. Dabei ist  $G(s)$  die Übertragungsfunktion und  $U(s)$  der Systemeingang im Bildbereich. Im Falle der Sprungantwort ist  $U(s) = \frac{1}{s}$  und somit der Grenzwert  $\lim_{s \rightarrow 0} G(s)$

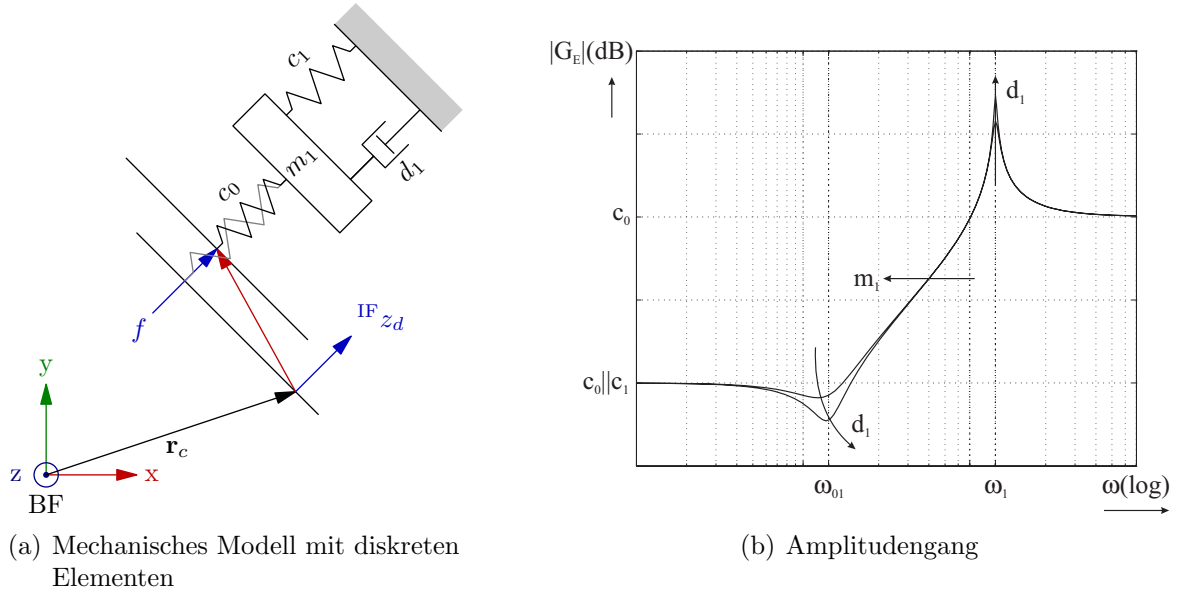


Abb. 4.4: Dynamisches Kontaktmodell

grund der Lageänderung  ${}^{\text{BF}}d\mathbf{x}$  berechnet. Die Zustandsraumdarstellung

$$\dot{\tilde{\mathbf{x}}} = \begin{bmatrix} 0 & 1 \\ -\frac{c_{e,0}+c_{e,1}}{m_{e,1}} & -\frac{d_{e,1}}{m_{e,1}} \end{bmatrix} \tilde{\mathbf{x}} + \begin{bmatrix} 0 \\ \frac{c_{e,0}}{m_{e,1}} \end{bmatrix} {}^{\text{IF}}z_d \quad (4.42a)$$

$${}^{\text{IF}}f_z = \begin{bmatrix} -c_{e,0} & 0 \end{bmatrix} \tilde{\mathbf{x}} + c_{e,0} \cdot {}^{\text{IF}}z_d \quad (4.42b)$$

der Übertragungsfunktion (4.38) beschreibt das Verhalten der Endeffektorkraft im Kontaktkoordinatensystem und im Zeitbereich. Dabei werden die neu eingeführten Zustandsgrößen durch  $\tilde{\mathbf{x}} \in \mathbb{R}^2$  repräsentiert und es gilt

$${}^{\text{IF}}\boldsymbol{\zeta} = \mathbf{s}^{\text{IF}} f_z \quad (4.43)$$

analog zu Gleichung (4.34). Die Endeffektorkräfte und -momente im Roboterkoordinatensystem BF können schließlich mit der Jacobimatrix (4.37) bestimmt werden.

Die durch das dynamische Kontaktmodell eingebrachte Nachgiebigkeit kann, im Gegensatz zu der oben angewandten Interpretation, auch dem Parallelroboter zugeordnet werden. Dann werden durch sie Elastizitäten der Struktur abgebildet. Auf die Synthese und das Verhalten der Kraft-Momentenregelung hat diese Interpretation keinen Einfluss.

Vergleichbare Modelle wurden in [Mills und Lokhorst, 1993] zur Kraftreglersynthese eingesetzt. Die Autoren modellieren damit die Systemdynamik im Umgebungskontakt *und* der Transitionsphase. Im Rahmen dieser Arbeit wird deshalb in der Transitionsphase eine dem unstetigen Verhalten des Systems angepasste Beschreibung verwendet.

## Systemdynamik

Befindet sich der Endeffektor oder ein fest mit diesem verbundenes Werkzeug im Umgebungskontakt, werden die Bewegungen des Parallelroboters durch die externen Kräfte und Momente eingeschränkt. Die bekannte Bewegungsgleichung (3.1) wird um den Term  $\mathbf{J}_x^{-T \text{ BF}} \mathbf{f}$  erweitert, der die externen Kräfte- und Momente vom kartesischen Roboterkoordinatensystem BF in Gelenkkordinaten transformiert. Das Systemverhalten im Umgebungskontakt wird durch

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{J}_x^{-T \text{ BF}} \mathbf{f} = \boldsymbol{\tau} \quad (4.44)$$

mit

$$\mathbf{f}^{\text{BF}} = \begin{cases} 0 : \Gamma(\mathbf{D}^T \mathbf{BF} \mathbf{T}_d(\mathbf{BF} \, d\mathbf{x}) \mathbf{t}) < 0 \\ \mathbf{BF} \mathbf{J}_{\text{IF}}(\mathbf{BF} \mathbf{x}_d) \mathbf{s}^{\text{IF}} f_z : \Gamma(\mathbf{D}^T \mathbf{BF} \mathbf{T}_d(\mathbf{BF} \, d\mathbf{x}) \mathbf{t}) \geq 0 \end{cases} \quad (4.37)$$

und

$$\mathbf{s}^{\text{IF}} f_z = g(\mathbf{s}^{\text{IF}} z_d) \quad (4.45)$$

$$\mathbf{x}_d^{\text{BF}} = \begin{bmatrix} \mathbf{D}^T \mathbf{BF} \mathbf{T}_d(\mathbf{BF} \, d\mathbf{x}) \mathbf{t} \\ \text{RPY}(\mathbf{D}^T \mathbf{BF} \mathbf{T}_d(\mathbf{BF} \, d\mathbf{x}) \mathbf{D}^T) \end{bmatrix} \quad (4.29)$$

$$\mathbf{s}^{\text{IF}} z_d = \mathbf{s}^T \mathbf{D}^T \mathbf{IF} \mathbf{T}_d(\mathbf{BF} \, d\mathbf{x}) \mathbf{t} \quad (4.33)$$

$$\mathbf{BF} \, d\mathbf{x} = \mathbf{h}(\mathbf{q}_1) - \mathbf{h}(\mathbf{q}_0) \quad (4.46)$$

beschrieben. Dabei ist  $\mathbf{J}_x^{-T}$  die Jacobimatrix der direkten Statik. Die Endeffektorkraft  $\mathbf{s}^{\text{IF}} f_z$  wird durch das statische (4.34) bzw. dynamische Kontaktmodell (4.42) bestimmt. Die Transformationen der Endeffektorbewegung wurden in den obigen Abschnitten hergeleitet. Durch die Lösung des direkten kinematischen Problems  $\mathbf{h}(\mathbf{q})$  wird die Lage des Endeffektors aus den Antriebswinkeln berechnet. Die Bedingung  $\Gamma \geq 0$  stellt sicher, dass sich das System im Umgebungskontakt befindet.

Alle Elemente und Transformation des Parallelroboters im Umgebungskontakt sind im Blockschaltbild 4.5 dargestellt. Das statische Kontaktmodell umfasst das nichtlineare Element der Totzone und die Umgebungssteifigkeit  $c_{e,0}$ . Ein zusätzliches PT<sub>2</sub>-Element im Rückführungszweig erweitert das Blockschaltbild zum dynamischen Kontaktmodell.

Die im Umgebungskontakt auftretenden externen Kräfte- und Momente können durch einen Kontaktregler, der den Regelkreis um die unterlagerte Antriebsregelung schließt, geregelt werden.



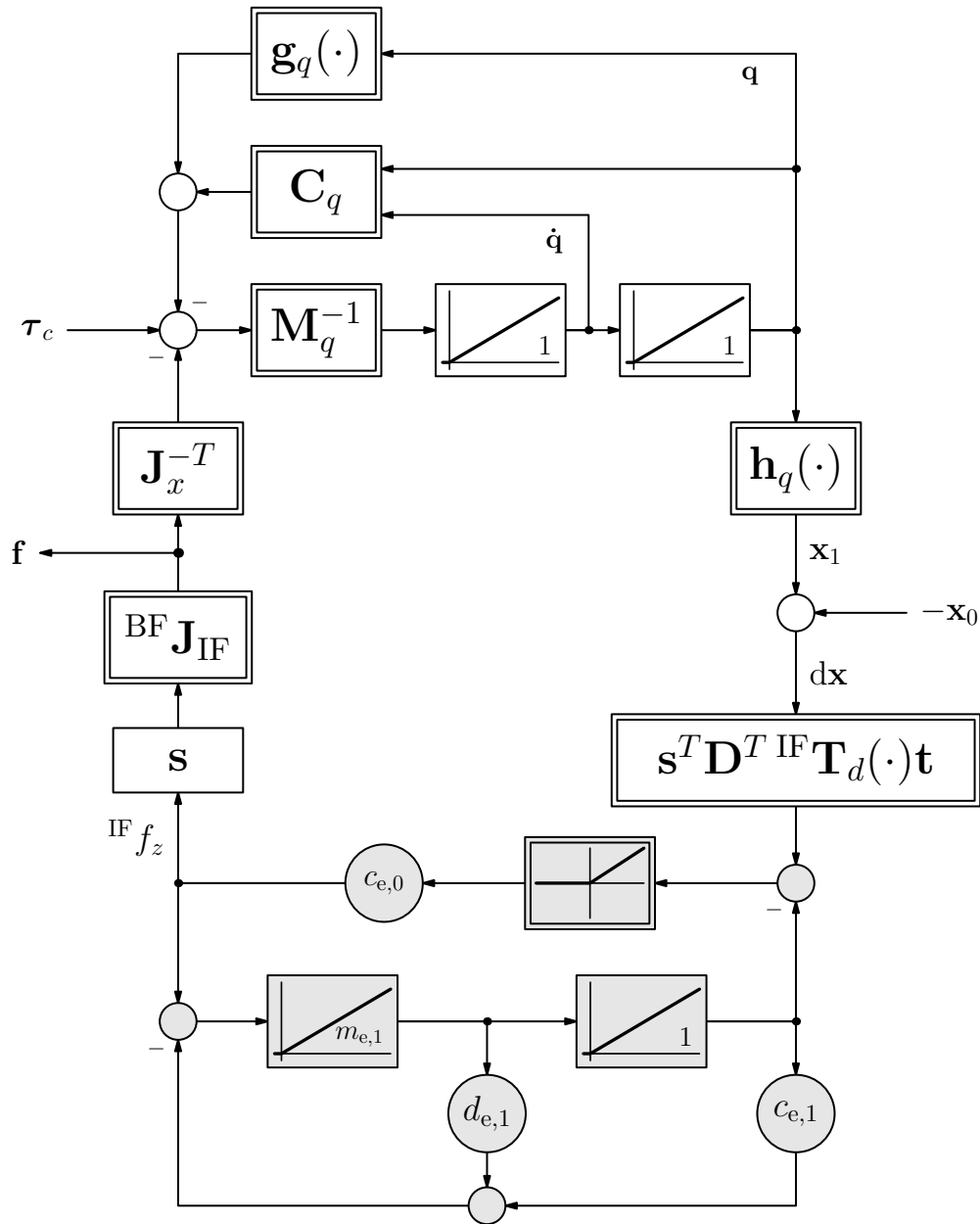


Abb. 4.5: Parallelroboter mit Transformationen und dynamischem Kontaktmodell. Die Elemente des Kontaktmodells sind grau hinterlegt.

### 4.3.2 Transitionsmodell

In der Transitionsphase nimmt der Endeffektor des Parallelroboters Kontakt mit der Umgebung auf. Bedingt durch Ungenauigkeiten in der Bahnplanung, Toleranzen in der Umgebungsgeometrie oder zu hohe Geschwindigkeiten unmittelbar vor der Kontaktaufnahme kann der Endeffektor von der Umgebungsgrenze abprallen und sich wieder in den Arbeitsraum zurückbewegen. Dieser Vorgang kann sich mehrmals wiederholen. Die Beschreibung des Systemverhaltens in dieser Phase durch Moreaus Stoßprozess führt auf eine differentielle Inklusion von Maßen [Kunze und Monteiro Marques, 2000].

#### Moreaus Stoßprozess

Bewegt sich der Parallelroboter im Freiraum, beschreibt die Gleichung

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^t \dot{\mathbf{x}}(\tau) d\tau \quad (4.47)$$

die Lage des Endeffektors. Der Vektor  $\mathbf{x}$  bezeichnet dabei die Lage im Roboterkoordinatensystem und  $\dot{\mathbf{x}}$  die entsprechenden Geschwindigkeiten. Der Arbeitsraum wird durch die Umgebung  $\partial\Phi$  begrenzt, so dass die Bewegung gemäß Gleichung (4.2) eingeschränkt wird und sich der Endeffektor zu jeder Zeit  $t$  im Raum

$$L = \left\{ \mathbf{x} \in \mathbb{R}^n : \Gamma_i(\mathbf{x}(t)) \leq 0 \quad \text{mit} \quad t \in [t_0; t_d] \right\} \quad (4.48)$$

befindet. In obiger Gleichung sind  $n$  die Freiheitsgrade des Parallelroboters. Die Umgebungsgeometrie  $\Gamma_i$  definiert Beschränkungen in den translatorischen Freiheitsgraden des kartesischen Roboterkoordinatensystems.

Im Falle eines Umgebungskontakts zum Zeitpunkt  $t_k$ , ändert sich die Geschwindigkeit  $\dot{\mathbf{x}}$  sprungförmig und die Beziehung in (4.47) kann nicht länger als wahr angenommen werden, da das Integral zum Zeitpunkt  $t_k$  keine endliche Lösung besitzt. Daher wird die *tatsächliche* Geschwindigkeit  $\dot{\mathbf{x}}$  durch eine Funktion beschränkter Variation  $\mathbf{u}$  ersetzt, so dass

$$\mathbf{x}(t) = \mathbf{x}(t_0) + \int_{t_0}^{t_d} \mathbf{u}(\tau) d\tau \quad (4.49)$$

gilt. Die Funktion besitzt einen rechten und linken Grenzwert zum Zeitpunkt des Umgebungskontakts  $t_k$ . Der linke Grenzwert von  $\mathbf{u}$  entspricht darüber hinaus der Annäherungsgeschwindigkeit  $\dot{\mathbf{x}}(t_k^-)$  während der rechte Grenzwert von  $\mathbf{u}$  die Geschwindigkeit nach dem Umgebungskontakt  $\dot{\mathbf{x}}(t_k^+)$  darstellt. Daraus folgt, dass  $\mathbf{u}$  Lebesgue-integrierbar ist und eine *generalisierte* Geschwindigkeit darstellt. Diese Vorüberlegungen führen die Maßtheorie zur Beschreibung des Systemverhaltens ein und ermöglichen so die Lösung der Differentialgleichungen im Umgebungskontakt [vgl. Brogliato, 1999, Kap. 1 und dort Anhang A-C].

Die Bewegungsgleichungen eines allgemeinen Euler-Lagrange-Systems lauten

$$\mathbf{M}(\mathbf{x}, t) \ddot{\mathbf{x}} - \mathbf{p}(\mathbf{x}, \dot{\mathbf{x}}, t) = \mathbf{r}(t). \quad (4.50)$$

Ein Vergleich mit dem dynamischen Modell des Parallelroboters in Gelenkkoordinaten (3.1) zeigt, dass  $-\mathbf{p}$  ein Vektorfeld der Gewichtskräfte  $\mathbf{g}(\mathbf{x})$  sowie der Coriolis- und Zentrifugalkräfte  $\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}})$  darstellt. Handelt es sich um ein geregeltes System, sind in  $\mathbf{p}$  ebenfalls die Stellgrößen enthalten. Die Matrix  $\mathbf{M}$  symbolisiert die bekannte Massenträgheitsmatrix. Anstatt der Antriebsmomente wird hier eine allgemeine Reaktion  $\mathbf{r}$  als Systemeingang angenommen. Unter Berücksichtigung der in Gleichung (4.49) eingeführten Funktion  $\mathbf{u}$  lautet die Bewegungsgleichung

$$\mathbf{r}(t) = \mathbf{M}(\mathbf{x}, t) \dot{\mathbf{u}} - \mathbf{p}(\mathbf{x}, \mathbf{u}, t) \quad \text{mit} \quad \dot{\mathbf{u}} = \ddot{\mathbf{x}}. \quad (4.51)$$

Da es sich bei  $\mathbf{u}$  um eine Funktion beschränkter Variation handelt, ist deren Ableitung ein Maß. Der Übergang von einer Darstellung der Parallelroboterdynamik im Freiraum als eine Gleichung von Funktionen in eine Gleichung von Maßen erfordert, dass die Reaktion  $\mathbf{r}$  ebenfalls ein Maß ist. Daraus folgt

$$d\mathbf{r} = \mathbf{M}(\mathbf{x}, t) d\mathbf{u} - \mathbf{p}(\mathbf{x}, \mathbf{u}, t) dt \quad (4.52)$$

wobei  $dt$  das Lebesgue-Maß auf dem Intervall  $[t_0; t_d]$  ist [Kunze und Monteiro Marques, 2000].

Im Freiraum ist die Reaktion null, d. h.  $d\mathbf{r} = \mathbf{0}$  und Gleichung (4.52) wird zu der bekannten, homogenen Differentialgleichung

$$\mathbf{M}(\mathbf{x}, t) \ddot{\mathbf{x}} - \mathbf{p}(\mathbf{x}, \dot{\mathbf{x}}, t) = \mathbf{0}.$$

Ändert sich die Geschwindigkeit sprunghaft, wie es zum Zeitpunkt des Umgebungskontakts  $t_k$  der Fall ist, ergibt sich eine andere Situation. An der Unstetigkeitsstelle  $t_k$  ist das Maß  $d\mathbf{u}$  gleichbedeutend mit dem Dirac-Impuls  $d\mathbf{u}|_{\{t_k\}} = [\mathbf{u}(t_k^+) - \mathbf{u}(t_k^-)] \delta_{t_k}$ . Da es sich bei  $\mathbf{u}$  um eine beschränkte Funktion handelt, ist das Stieltjes-Maß von  $\{t_k\}$

$$d\mathbf{u}(\{t_k\}) = \int_{\{t_k\}} d\mathbf{u} = \mathbf{u}(t_k^+) - \mathbf{u}(t_k^-). \quad (4.53)$$

Gleichzeitig ist das Lebesgue-Maß von  $\{t_k\}$  null, so dass

$$\int_{\{t_k\}} \mathbf{p}(\mathbf{x}, \dot{\mathbf{x}}, t) dt = 0 \quad (4.54)$$

wird. Durch Einsetzen der Gleichungen (4.53) und (4.54) in die Systemgleichung (4.52) kann schließlich die Reaktion  $d\mathbf{r}$  als Differenz der Annäherungsgeschwindigkeit  $\mathbf{u}(t_k^-)$  und

Abprallgeschwindigkeit  $\mathbf{u}(t_k^+)$  dargestellt werden

$$d\mathbf{r}(\{t_k\}) = \mathbf{M}(\mathbf{x}, t_k) d\mathbf{u}(\{t_k\}) = \mathbf{M}(\mathbf{x}, t_k) [\mathbf{u}(t_k^+) - \mathbf{u}(t_k^-)]. \quad (4.55)$$

Die Systemreaktion beim Aufprall des Endeffektors auf die Umgebung wird ausschließlich durch die Massenträgheitsmatrix  $\mathbf{M}$  bestimmt. Stellgrößen des Reglers im Vektorfeld  $\mathbf{p}$  sind wirkungslos.

In der Beschreibung des Parallelroboters im Umgebungskontakt durch den Stoßprozess von Moreau sind die generalisierten Geschwindigkeiten die unbekannten Größen. Die Berechnung der linken und rechten Ableitung im Umgebungskontakt bei  $\Gamma_i(\mathbf{x}(t_k)) = 0$  liefert die folgenden Bedingungen

$$\langle u(t_k^+), \nabla \Gamma_i(\mathbf{x}(t_k)) \rangle \leq 0 \quad \text{und} \quad \langle u(t_k^-), \nabla \Gamma_i(\mathbf{x}(t_k)) \rangle \geq 0 \quad \text{mit} \quad i \in [1; m]_{\mathbb{N}},$$

denen die Geschwindigkeiten gehorchen müssen. Daraus folgt, dass

$$\mathbf{u}(t_k^+) \in V(\mathbf{x}(t)) \quad \text{und} \quad -\mathbf{u}(t_k^-) \in V(\mathbf{x}(t)) \quad (4.56)$$

wobei

$$V(\mathbf{x}(t)) := \begin{cases} \left\{ \mathbf{w} \in \mathbb{R}^n : \langle \mathbf{w}, \nabla \Gamma_i(\mathbf{x}) \rangle \leq 0 \right\} : \Gamma_i(\mathbf{x}) \geq 0 \\ \mathbb{R}^n : \Gamma_i(\mathbf{x}) < 0 \end{cases} \quad (4.57)$$

der Tangentialkegel zum Arbeitsraum  $L$  am Punkt  $\mathbf{x}$  ist. Im Falle eines reibungsfreien Kontakts ist die Reaktion entlang des Normalenkegels

$$N_{V(\mathbf{x}(t))}(\mathbf{u}) := \begin{cases} \mathbf{0} : \Gamma_i(\mathbf{x}(t)) > 0 \\ \left\{ \mathbf{a} : \langle \mathbf{a}, \mathbf{v} - \mathbf{u} \rangle \leq 0 \forall \mathbf{v} \in V(\mathbf{x}(t)) \right\} : \Gamma_i(\mathbf{x}(t)) = 0 \end{cases} \quad (4.58)$$

gerichtet, der wiederum senkrecht auf dem Tangentialkegel  $V(\mathbf{x}(t))$  steht. Das heißt die Reaktion  $d\mathbf{r}$  findet in dem Geschwindigkeitsraum

$$-d\mathbf{r} \in N_{V(\mathbf{x}(t))}(\mathbf{u}(t)) \quad (4.59)$$

statt. Zusammengefasst mit der oben entwickelten Bewegungsgleichung (4.52) gilt schließlich

$$-d\mathbf{r} = \mathbf{p}(\mathbf{x}, \mathbf{u}, t) dt - \mathbf{M}(\mathbf{x}, t) d\mathbf{u} \in N_{V(\mathbf{x}(t))}(\mathbf{u}(t)). \quad (4.60)$$

Die Berechnungen bis zu diesem Punkt liefern als wesentliches Ergebnis die allgemeingültige Beschreibung der mechanischen Vorgänge während eines Stoßes. Die Amplitude der Reaktion

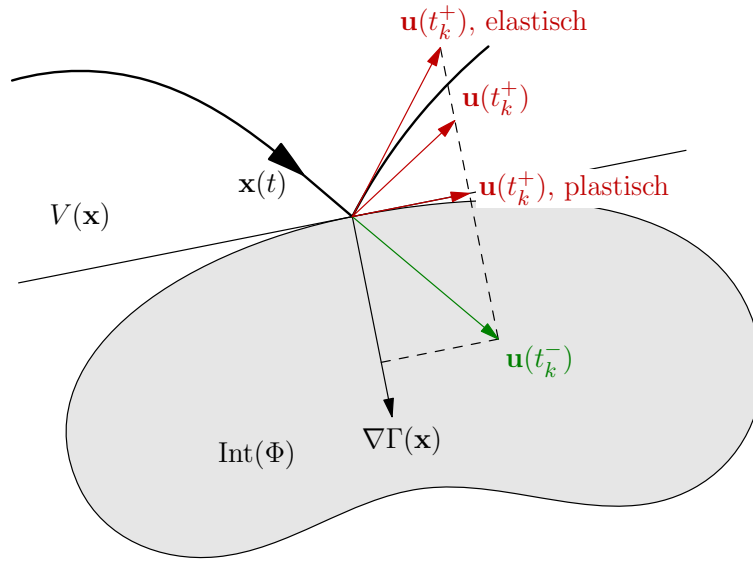


Abb. 4.6: Kollision des Endeffektors mit der Umgebung modelliert mit Moreaus Stoßprozess. Dargestellt ist die Trajektorie  $\mathbf{x}(t)$  und die dazugehörige Annäherungs-  $\mathbf{u}(t_k^-)$  sowie die Abprallgeschwindigkeit  $\mathbf{u}(t_k^+)$  für verschiedene Stoßzahlen.

$d\mathbf{r}$  wird in Gleichungen (4.55) beschrieben, während Gleichung (4.60) die Richtung der externen Kräfte beschreibt.

Die generalisierte Geschwindigkeit  $\mathbf{u}(t_k)$  wird als ein gewichteter Durchschnitt aus der Annäherungs- und Abprallgeschwindigkeit eingeführt

$$\mathbf{u}(t_k) = \frac{1}{1+e} \mathbf{u}(t_k^+) + \frac{e}{1+e} \mathbf{u}(t_k^-). \quad (4.61)$$

Dabei ist  $e$  die Newtonsche Stoßzahl, die das Verhältnis von Annäherungs- und Abprallgeschwindigkeit beschreibt. Für einen plastischen Stoß ist  $e = 0$  und ein elastischer Stoß wird durch  $e = 1$  beschrieben. Da der Umgebungskontakt des Endeffektors immer einen Abbau der kinetischen Energie zur Folge hat, ist  $e < 1$  und das Transitionsmodell enthält dadurch einen dissipativen Term [vgl. Pagilla, 2001; Pagilla und Biao, 2001]. Der Wert der Stoßzahl  $e$  für eine reale Umgebung kann durch Versuche ermittelt werden.

In Abbildung 4.6 ist die Kollision des Endeffektors mit der Umgebung dargestellt. Die Annäherungsgeschwindigkeit  $\mathbf{u}(t_k^-)$  wird am Tangentialkegel  $V(\mathbf{x})$  gespiegelt. Es ist ein plastischer, teilelastischer und elastischer Stoß dargestellt. Die Abprallgeschwindigkeit  $\mathbf{u}(t_k^+)$  mit Winkel und Wert wird durch die Stoßzahl  $e$  bestimmt.

Die Zusammenfassung der Herleitung nach [Kunze und Monteiro Marques, 2000] führt auf die Formulierung des Stoßprozesses von Moreau:

**Problem 1 (Moreaus Stoßprozess)** Gegeben ist der Anfangszustand des Systems nach Gleichung (4.51) mit  $\mathbf{x}_0 \in L$  und  $\mathbf{u}_0 \in V(\mathbf{x}_0)$  sowie dem Vektorfeld  $\mathbf{p}$  und der Massenträgheitsmatrix  $\mathbf{M}$ . Gesucht wird die Funktion beschränkter Variation  $\mathbf{u}$  mit  $\mathbf{u}(0) = \mathbf{u}_0$ , so dass

gilt

- $\mathbf{x}(t) = \mathbf{x}_{t_0} + \int_{t_0}^{t_d} \mathbf{u}(\tau) d\tau,$
- $\mathbf{u}(t) \in V(\mathbf{x}(t)) \quad \forall t \in [t_0; t_d]$  und
- $\mathbf{p}(\mathbf{x}, \mathbf{u}, t)dt - \mathbf{M}(\mathbf{x}, t)d\mathbf{u} \in N_{V(\mathbf{x}(t))}(\mathbf{u}(t)).$

Weiterhin existiert auf dem Intervall  $[t_0; t_d]$  ein positives Maß  $d\mu$  bezüglich dessen das Lebesgue-Maß  $dt$  und das Stieltjes-Maß  $d\mathbf{u}$  eine Dichte besitzen, d. h.  $t'_\mu = \frac{dt}{d\mu}$  und  $\mathbf{u}'_\mu = \frac{d\mathbf{u}}{d\mu}$ , so dass

$$\mathbf{p}(\mathbf{x}, \mathbf{u}, t)t'_\mu - \mathbf{M}(\mathbf{x}, t)\mathbf{u}'_\mu \in N_{V(\mathbf{x}(t))}(\mathbf{u}(t)). \quad (4.62)$$

Dann ist  $\mathbf{x}(t)$  die Bahn und  $\mathbf{u}(t)$  die zugehörige Geschwindigkeit des Systems in der Transitionsphase.

### Diskretisierung

Die Lösung des Stoßprozesses von Moreau beinhaltet eine Diskretisierung und berücksichtigt daher implizit Aspekte der Implementierung. Ausgehend von dem folgenden Theorem wird der Algorithmus entwickelt.

**Theorem 1 (Lösung des Stoßprozesses von Moreau)** Gegeben sei eine glatte Umgebungsfunktion  $\Gamma_i(\mathbf{x})$  mit  $\nabla \Gamma_i(\mathbf{x}) \neq \mathbf{0}$  und einem kontinuierlichen und beschränkten Vektorfeld  $\mathbf{p}$ . Dann gelten folgende Annahmen:

- (a) Die zu konstruierende Folge  $(\mathbf{u}_v, \mathbf{x}_v)$  konvergiert gleichmäßig gegen einen Grenzwert  $(\mathbf{u}, \mathbf{x})$ , für den  $\mathbf{u}(t) \in V(\mathbf{x}(t))$  für alle  $t \in [t_0; t_d]$  gilt.
- (b) Die generalisierte Geschwindigkeit  $\mathbf{u}(t_k)$  aus Gleichung (4.61) ist eine Lösung des Stoßprozesses von Moreau aus Problem 1.

Die Schrittweite der Diskretisierung sei  $h = T/v$ , wobei  $v$  die Anzahl der Schritte angibt und die endlichen Folgen  $(\mathbf{x}_i^v)$ ,  $(\mathbf{u}_i^v)$  sowie  $(\mathbf{w}_i^v)$  für  $i \in [0; v]_{\mathbb{N}}$  definiert. Deren Startwerte lauten  $\mathbf{x}_0^v = \mathbf{x}_0$  und  $\mathbf{u}_0^v = \mathbf{w}_0^v = \mathbf{u}_0$ . Die diskrete Integration der generalisierten Geschwindigkeit nach Gleichung (4.49) ergibt die Lage

$$\mathbf{x}_{i+1}^v = \mathbf{x}_i^v + (t_{i+1}^v - t_i^v) \mathbf{u}_i^v. \quad (4.63)$$

Weiterhin wird die Bewegungsgleichung im Umgebungskontakt (4.52) normiert, so dass

$$\tilde{\mathbf{p}} = \mathbf{M}^{-1}\mathbf{p} \quad (4.64)$$

ist und schließlich als

$$(t_{i+1}^v - t_i^v) \tilde{\mathbf{p}}(\mathbf{x}_{i+1}^v, t_{i+1}^v) - (\mathbf{u}_{i+1}^v - \mathbf{u}_i^v) \in N_{V(\mathbf{x}_{i+1}^v)}(\mathbf{w}_{i+1}^v) \quad (4.65)$$

diskretisiert werden kann. Dieses Vorgehen ist möglich, da die Massenträgheitsmatrix  $\mathbf{M}$  positiv definit und damit zu invertieren ist. Durch die Definition von  $k := \frac{e}{1+e}$  kann die generalisierte Geschwindigkeit (4.61) als

$$\mathbf{w}_{i+1}^v = k\mathbf{u}_i^v + (1-k)\mathbf{u}_{i+1}^v \quad (4.66)$$

notiert werden. In Gleichung (4.65) wird der Tangentialkegel  $V$  sowie das Vektorfeld  $\tilde{\mathbf{p}}$  an der Endeffektorlage  $\mathbf{x}_{i+1}^v$  berechnet. Für  $v \rightarrow \infty$  stellen die Glieder  $\mathbf{u}_i^v$  und  $\mathbf{u}_{i+1}^v$  den linken  $\mathbf{u}(t_k^-)$  bzw. rechten Grenzwert  $\mathbf{u}(t_k^+)$  der generalisierten Geschwindigkeit dar. Da es sich bei  $N$  um einen Kegel handelt, kann die linke Seite der Gleichung mit dem positiven Faktor  $(1-k)$  multipliziert werden ohne, dass sich die Aussage ändert

$$(1-k)h\tilde{\mathbf{p}}(\mathbf{x}_{i+1}^v, t_{i+1}^v) - (1-k)(\mathbf{u}_{i+1}^v - \mathbf{u}_i^v) \in N_{V(\mathbf{x}_{i+1}^v)}(\mathbf{w}_{i+1}^v). \quad (4.67)$$

Die folgende Beziehung aus der konvexen Analysis stellt das Herzstück der Diskretisierung von Moreaus Stoßprozess dar und ermöglicht die numerische Integration:

**Lemma 1 (Projektion [Kunze und Monteiro Marques, 2000])** *Gegeben ist eine geschlossene und konvexe Kurve  $C$  und ein beliebiger Punkt  $\mathbf{x}$ . Dann stellt die Projektion  $\mathbf{y} = \text{proj}(\mathbf{x}, C)$  die geometrisch kürzeste Entfernung zwischen  $\mathbf{x}$  und der Kurve  $C$  dar. Darüber hinaus gilt*

$$\mathbf{y} = \text{proj}(\mathbf{x}, C) \quad \text{genau dann, wenn} \quad \mathbf{y} \in C, \mathbf{x} - \mathbf{y} \in N_C(y) \quad \text{ist.}$$

Durch die Anwendung dieser Projektion auf Gleichung (4.67) kann der gewichtete Durchschnitt der generalisierten Geschwindigkeit mit

$$\mathbf{w}_{i+1}^v = \text{proj}\left(\mathbf{u}_i^v + (1-k)h\tilde{\mathbf{p}}(\mathbf{x}_{i+1}^v, t_{i+1}^v), V(\mathbf{x}_{i+1}^v)\right) \quad (4.68)$$

bestimmt werden. Durch die Umkehrfunktion von Gleichung (4.61) wird schließlich die Abprallgeschwindigkeit  $\mathbf{u}_{i+1}^v$  aus der Annäherungsgeschwindigkeit  $\mathbf{u}_i^v$  und der Projektion berechnet.

Der Algorithmus zur Berechnung der Endeffektortrajektorie mit Hilfe von Moreaus Stoßprozess wird *Catching-up-Algorithmus* genannt und lässt sich wie folgt zusammenfassen:

- 1.) Aktualisieren der Lage  $\mathbf{x}_{i+1}^v$  nach Gleichung (4.63).
- 2.) Berechnung der Projektion  $\mathbf{w}_{i+1}^v$  nach Gleichung (4.68).

3.) Bestimmung der neuen Geschwindigkeit durch

$$\mathbf{u}_{i+1}^v = \frac{\mathbf{w}_{i+1}^v - k\mathbf{u}_i^v}{1 - k}. \quad (4.69)$$

Kunze und Monteiro Marques [2000] zeigen, dass für  $v \rightarrow \infty$  die Folge  $(\mathbf{u}_v, \mathbf{x}_v)$  gegen einen Grenzwert  $(\mathbf{u}, \mathbf{x})$  strebt und die generalisierte Geschwindigkeit (4.61) eine Lösung des Stoßprozesses ist.

### Simulation

Im Folgenden wird die Implementierung von Moreaus Stoßprozess zur Simulation eines einfachen mechanischen Systems im Umgebungskontakt beschrieben. Das Szenario bildet ein Ball, der aufgrund der Gravitationskraft auf eine Oberfläche fällt und dort abprallt. Die Bewegungsgleichung des Euler-Lagrange-Systems nach Gleichung (4.50) lautet in diesem Fall

$$\ddot{x} = g + r. \quad (4.70)$$

In obiger Gleichung ist  $x$  die Ortsvariable,  $g$  die Erdbeschleunigung, die in die negative  $x$ -Richtung zeigt und  $r$  die Reaktion. Der unilaterale Umgebungskontakt sei durch

$$L = \{x \in \mathbb{R} : -x \leq 0\} \quad (4.71)$$

beschrieben. Der entsprechende Tangentialkegel ist

$$V(x) = \begin{cases} \{w \in \mathbb{R} : -x \cdot w \leq 0\} & : x \leq 0 \\ \mathbb{R} & : x > 0 \end{cases} \quad (4.72)$$

und beschreibt die Halbgerade über der Umgebungsgrenze. Zur Ausführung des *Catching-up-Algorithmus* muss die Projektion in Gleichung (4.68), d. h. das Element  $w_{i+1}^v \in V(x_{i+1})$ , das den Abstand zwischen  $u_i^v + (1-k)hg$  und  $w_{i+1}^v$  minimiert, bestimmt werden. Im Freiraum, für den Fall  $x > 0$ , ist  $w_{i+1}^v = u_i^v + (1-k)hg$ , so dass der Abstand null wird. Im Umgebungskontakt ist  $x \leq 0$  und  $w_{i+1}^v = 0$ . Der Abstand beträgt dann mindestens  $u_i^v + (1-k)hg$ . Zusammengefasst ergibt sich

$$w_{i+1}^v = \begin{cases} 0 & : x \leq 0, \\ u_i^v + (1-k)hg & : x > 0. \end{cases} \quad (4.73)$$

Für eine gegebene Stoßzahl  $e = 0,8$  kann  $k = \frac{e}{1+e}$  nach Gleichung (4.66) berechnet werden. Mit der Vorgabe einer Schrittweite von  $h = 10 \mu\text{s}$  sind alle Voraussetzungen für die Simulation des Stoßprozesses geschaffen, deren Ergebnisse die Abbildung 4.7 zeigt. Kurve (a) stellt



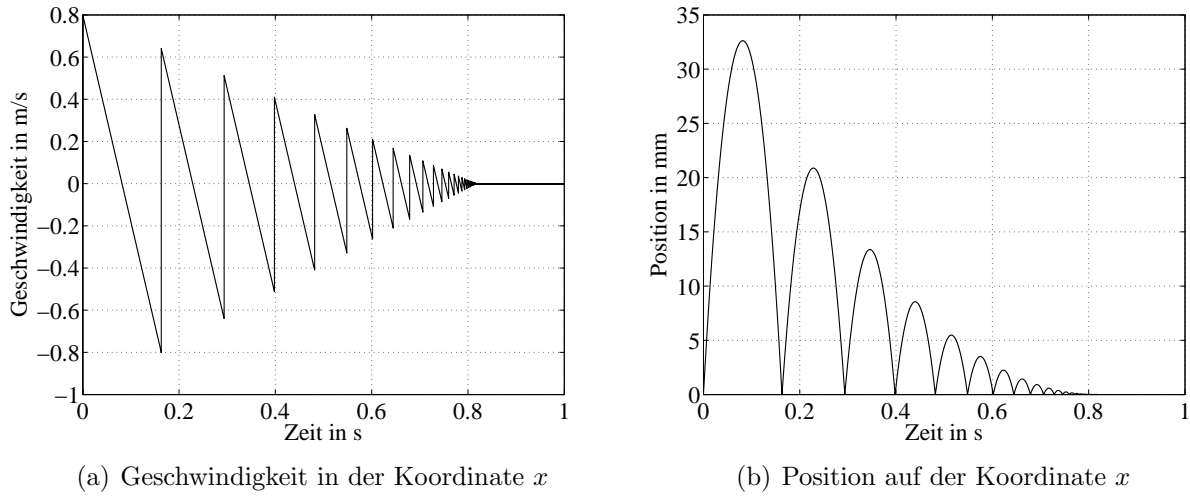


Abb. 4.7: Simulationsergebnisse von Moreaus Stoßprozess. Ein Ball fällt durch Einwirkung der Gewichtskraft auf eine Oberfläche.

die Geschwindigkeit dar, während Kurve (b) die Position zeigt. Der Ball prallt mit einer Anfangsgeschwindigkeit von  $\dot{x} = -1 \text{ m/s}$  auf die Oberfläche. Aufgrund des teilelastischen Stoßes wird ein Teil der Energie absorbiert. Der Ball verlässt die Oberfläche mit  $\dot{x} = 0,8 \text{ m/s}$  und erreicht anschließend eine Höhe von ca.  $x = 33 \text{ mm}$ . Da keinerlei Verluste wie Luftreibung oder andere externe Kräfte wirken, erreicht der Ball die Oberfläche wieder mit der gleichen Geschwindigkeit, mit der er von ihr abgeprallt ist. Nach  $0,82 \text{ s}$  sind die transienten Vorgänge abgeschlossen und der Ball ruht auf der Oberfläche.

### Eigenschaften des Stoßprozesses

Der Stoßprozess von Moreau beschreibt das Verhalten des Parallelroboters während der Transitionsphase in seiner Gesamtheit. In der Entwicklung des Verfahrens wurden mathematische, mechanische und numerische Aspekte berücksichtigt: Durch die Beschreibung der Dynamik als eine Gleichung von Maßen sind keine Annahmen über das Systemverhalten im Umgebungskontakt notwendig [vgl. Brogliato, 1999, S. 4]. Die Ergebnisse der Berechnungen führen direkt auf die Beschreibung des Stoßes als ein Prozess, bei dem sich in infinitesimal kurzer Zeit die Geschwindigkeit sprunghaft ändert, während die Lage des Objekts konstant bleibt. Die Richtung der auftretenden Kräfte wird von der Umgebung bestimmt und ist parallel zum Normalenvektor gerichtet.

Der durch die Diskretisierung entwickelte Catching-up-Algorithmus ist im Freiraum identisch mit dem impliziten Eulerverfahren zur Lösung von Differentialgleichungen. Im Kontaktfall ergeben sich allerdings signifikante Unterschiede. Das Eulerverfahren basiert auf der Annäherung der Geschwindigkeit  $\dot{\mathbf{x}}$ , die an der Unstetigkeitsstelle  $t_k$  nicht existiert. Im Gegensatz dazu berechnet der Catching-up-Algorithmus das immer definierte Maß  $d\mathbf{u}$  [vgl.

Acary und Brogliato, 2005, Abschnitt 2.1].

Die Berechnung von Moreaus Stoßprozess führt bei sehr kleiner Schrittweite auf die linken und rechten Grenzwerte der Geschwindigkeit an der Unstetigkeitsstelle [vgl. Kunze und Monteiro Marques, 2000, S. 44]. Das System wird daher bei jedem Stoß neu initialisiert, ohne dass numerische Ungenauigkeiten zu Energieverlusten bzw. -zuwächsen führen. Weiterhin sei angemerkt, dass mit dem Algorithmus multilaterale Kontaktsituationen simuliert werden können.

Ein Transitionsplaner beeinflusst das Systemverhalten zwischen den Kontakten des Parallelroboters mit der Umgebung so, dass der Endeffektor nach einer endlichen Zeit auf der Oberfläche zur Ruhe kommt. Der in diesem Abschnitt hergeleitete Stoßprozess von Moreau kann zur Validierung des Algorithmus durch Simulationen herangezogen werden.

# Kapitel 5

## Kontaktregelung

DIE Kontaktregelung ist die Grundlage zur Ausführung von Montageaufgaben mit Parallelrobotern. Durch geeignete Bewegungsbefehle des Aktionsprimitivnetzes wird der Endeffektor mit dem daran befestigten Werkzeug nahe an die Umgebung herangeführt, ohne dass er die Oberfläche berührt. Die anschließend auszuführende Kontaktregelung sorgt für eine sichere Kontaktaufnahme und dafür, dass die gewünschte Endeffektorkraft bzw. das gewünschte Endeffektormoment aufgebaut wird. Dieser Ansatz berücksichtigt die in Brogliato [1999] explizit beschriebene Einteilung der Kontaktregelung in die drei Phasen der Freiraumbewegung, der Transition und der Kraft-Momentenregelung.

### 5.1 Kontaktregelungskonzept

Das in dieser Arbeit vorgestellte Kontaktregelungskonzept umfasst einen Planungs- bzw. Regelungsalgorithmus für die Transitionsphase und die Kraft-Momentenregelung unter Berücksichtigung der in Kapitel 2 beschriebenen Anforderungen einer Robotersteuerung auf der Basis von Aktionprimitiven. Die Steuerung der Freiraumbewegung wird nicht näher behandelt, sondern als gegeben vorausgesetzt.

Gemäß der Definition in Abschnitt 2.2.1 handelt es sich bei der Kontaktregelung um ein Bewegungsmodul. In Abbildung 5.1 ist das Blockschaltbild der Kontaktregelung dargestellt. Auf der linken Seite befinden sich die Elemente des Bewegungsmoduls im Task-Frame. Diese sind der Regelungsalgorithmus in allgemeiner Darstellung  $\mathbf{G}_{K_3}$  und ein Führungsgrößengenerator bestehend aus drei Integratoren mit proportionalen Elementen  $\mathbf{a}_i$  in der Rückführung. Desweiteren ist ein Filter der Vorsteuergröße  $\mathbf{G}_{\text{fwd}}$  dargestellt.

Im Roboterkoordinatensystem auf der rechten Seite sind die Elemente des inversen dynamischen Modells und der unterlagerten Antriebsregelung zu erkennen. Die Systemdynamik im Umgebungskontakt ist durch das entsprechende Modell angedeutet. Die Endeffektorkräfte bzw. -momente werden auf den Eingang des Parallelroboters zurückgeführt und an den Task-Frame-Formalismus zur Transformation in den Task-Frame übergeben.

In den folgenden Abschnitten wird die Auslegung der oben beschriebenen Elemente diskutiert und so die Implementierung der Kontaktregelung für Parallelroboter entwickelt.

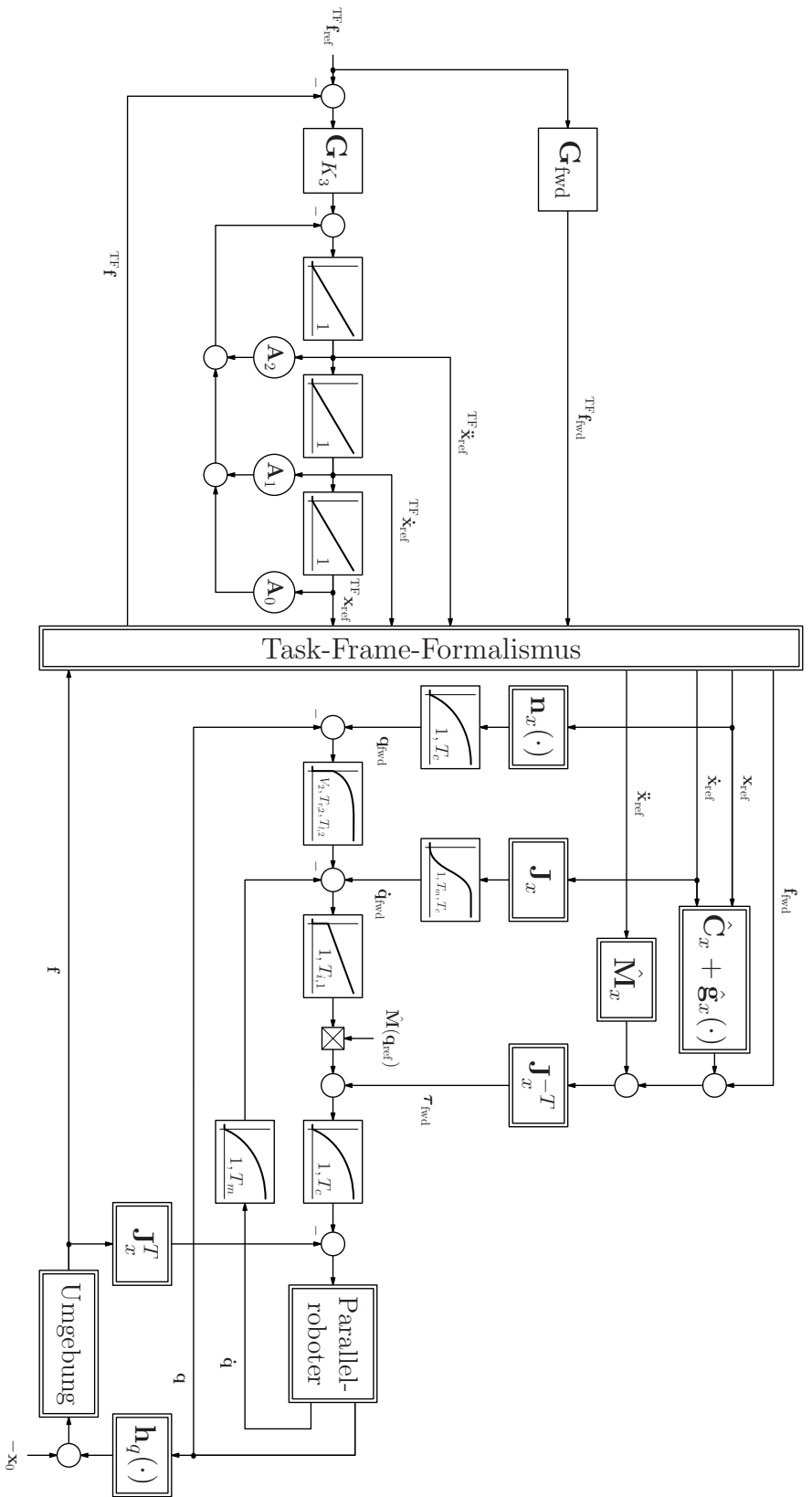


Abb. 5.1: Blockschaltbild der Kraft-Momentenregelung im Task-Frame mit unterlagerter Antriebsregelung und Umgebungsmodell im Roboterkoordinatensystem

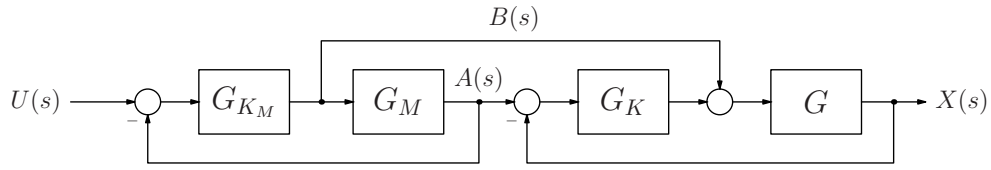


Abb. 5.2: Modellfolgeregelung

### 5.1.1 Modellfolgeregelung

Das Konzept der Kraft-Momentenregelung fußt auf den Grundlagen der in Abbildung 5.2 dargestellten Modellfolgeregelung. Bei dieser Art der Regelung ist dem Regelkreis, bestehend aus der Regelstrecke  $G$  und einem Regler  $G_K$  ein zusätzlicher Regelkreis vorgeschaltet. Letzterer besteht aus einem Modell der Regelstrecke  $G_M$  und einem Regler  $G_{K_M}$ , der auf ein optimales Führungsverhalten ausgelegt ist. Die Stellgrößen des Modellreglers sind gleichzeitig die Führungsgrößen des eigentlichen Regelkreises. Die Übertragungsfunktion der Modellfolgeregelung leitet sich folgendermaßen ab: Die internen Signale der Führungsgröße und Vorsteuerung sind

$$A(s) = \frac{G_{K_M} G_M}{1 + G_{K_M} G_M} U(s) \quad \text{bzw.} \quad (5.1a)$$

$$B(s) = \frac{G_{K_M}}{1 + G_{K_M} G_M} U(s). \quad (5.1b)$$

Die Gleichung für das Ausgangssignal lautet

$$X(s) = \frac{G_K G}{1 + G_K G} A(s) + \frac{G}{1 + G_K G} B(s). \quad (5.1c)$$

Daraus folgt schließlich die Übertragungsfunktion

$$\frac{X(s)}{U(s)} = \frac{G_{K_M} G (1 + G_K G_M)}{(1 + G_K G) (1 + G_{K_M} G_M)}. \quad (5.2)$$

Die im Allgemeinen betrachtete Anwendung geht davon aus, dass das Modell der Regelstrecke  $G_M$  in ausreichender Genauigkeit bekannt ist. Dann folgt die Regelstrecke  $G$  den Führungsgrößen und der Regler  $G_K$  muss nur beim Auftreten externer Störungen und Modellfehlern eingreifen. Wie die Übertragungsfunktion

$$\frac{X(s)}{U(s)} = \frac{G_{K_M} G}{1 + G_{K_M} G} \quad (5.3)$$

zeigt, verhält sich das System als würde die Strecke  $G$  ausschließlich von dem Modellregler  $G_{K_M}$  angesteuert werden. Der für das Konzept der Kontaktregelung ausschlaggebende Fall weicht von dieser Betrachtungsweise ab. Er geht davon aus, dass das Modell *und* der

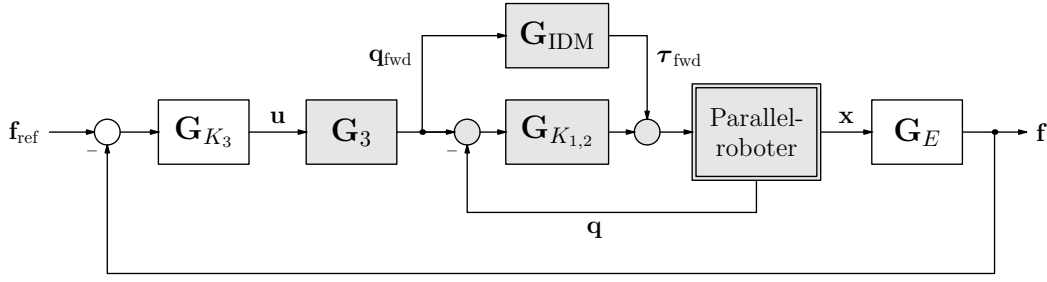


Abb. 5.3: Modellfolgeregelungskonzept der Kontaktregelung. Die Elemente der Modellfolgeregelung sind grau hinterlegt.

Modellregler in ausreichender Genauigkeit mit der Regelstrecke bzw. dem Regler  $G_K$  übereinstimmen. Die dann gültige Übertragungsfunktion

$$\frac{X(s)}{U(s)} = \frac{G_K G}{1 + G_K G} \quad (5.4)$$

zeigt, dass das Systemverhalten dem ohne vorgeschalteten Modellregelkreis entspricht.

Im Fall der Kontaktregelung bildet die in Kapitel 3 beschriebene unterlagerte Antriebsregelung, zusammen mit einem noch zu definierenden Führungsgrößengenerator, die beiden Regelkreise einer Modellfolgeregelung. Durch das Umgebungsmodell und einen Kontaktregler wird um diese ein äußerer Regelkreis geschlossen. Das Blockschaltbild ist in Abbildung 5.3 dargestellt. Die Transformationen der Führungs- und Vorsteuergrößen sind zur Wahrung der Übersichtlichkeit weggelassen. Der Block  $G_3$  repräsentiert den Führungsgrößengenerator und  $G_{K3}$  die Kontaktregelung. Das inverse dynamische Modell des Parallelroboters aus Gleichung (3.14) ist durch  $G_{IDM}$  dargestellt, während  $G_{K1,2}$  die Kaskadenregelung (3.10) des Parallelroboters  $G$  symbolisiert. Dann kann das Konzept der Kraft-Momentenregelung wie folgt definiert werden:

**Korollar 3 (Modellfolgeregelung)** *Bildet der Führungsgrößengenerator  $G_3$  die unterlagerte Antriebsregelung exakt nach, d. h.*

$$G_3 := G G_{K1,2} (I + G G_{K1,2})^{-1}, \quad (5.5)$$

*und entspricht das inverse dynamische Modell dem inversen Modell des Parallelroboters*

$$G_{IDM} := G^{-1}, \quad (5.6)$$

*dann bilden beide Übertragungsfunktionen eine Modellfolgeregelung nach und der geschlossene Regelkreis der Kraft-Momentenregelung von der Sollkraft  $f_{ref}$  zur Endeffektorkraft  $f$  hat die Übertragungsfunktion*

$$G_{FT} = G_E G_3 G_{K3} (I + G_E G_3 G_{K3})^{-1}. \quad (5.7)$$

In dem folgenden Beweis wird gezeigt, dass bei geeigneter Wahl der Parameter des Führungsgrößengenerators von diesem keine weitere Verzögerung in den Regelkreis eingebracht wird.

BEWEIS (MODELLFOLGEREGELUNG) Ohne Berücksichtigung der Transformationen zwischen dem Roboterkoordinatensystem der unterlagerten Antriebsregelung und dem Task-Frame der Bewegungsmodule ist die Übertragungsfunktion vom Eingang des Führungsgrößengenerators zum Ausgang des Parallelroboters

$$\mathbf{X}(s) = \mathbf{G}\mathbf{G}_{K_{1,2}} (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-1} \mathbf{Q}_{\text{fwd}}(s) + \mathbf{G} (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-1} \mathbf{T}_{\text{fwd}}(s). \quad (5.8)$$

Die Bestimmungsgleichung des Führungsgrößengenerators (5.5) führt auf

$$\mathbf{Q}_{\text{fwd}}(s) = \mathbf{G}_3 \mathbf{U}(s) = \mathbf{G}\mathbf{G}_{K_{1,2}} (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-1} \mathbf{U}(s). \quad (5.9)$$

Die Vorsteuersignale, in diesem Fall die Drehmomente  $\mathbf{T}_{\text{fwd}}(s)$ , lauten

$$\mathbf{T}_{\text{fwd}}(s) = \mathbf{G}_{\text{IDM}} \mathbf{G}_3 \mathbf{U}(s) = \mathbf{G}\mathbf{G}_{K_{1,2}} \mathbf{G}_{\text{IDM}} (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-1} \mathbf{U}(s). \quad (5.10)$$

Dann ist Gleichung (5.8)

$$\begin{aligned} \mathbf{X}(s) &= \left[ \mathbf{G}^2 \mathbf{G}_{K_{1,2}} \mathbf{G}_{\text{IDM}} + (\mathbf{G}\mathbf{G}_{K_{1,2}})^2 \right] (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-2} \mathbf{U}(s) \\ &= \mathbf{G}\mathbf{G}_{K_{1,2}} [\mathbf{G}\mathbf{G}_{\text{IDM}} + \mathbf{G}\mathbf{G}_{K_{1,2}}] (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-2} \mathbf{U}(s). \end{aligned}$$

Unter Berücksichtigung der Definition des inversen dynamischen Modells aus Gleichung (5.6) vereinfacht sich obige Gleichung zu

$$\mathbf{X}(s) = \mathbf{G}\mathbf{G}_{K_{1,2}} (\mathbf{I} + \mathbf{G}\mathbf{G}_{K_{1,2}})^{-1} \mathbf{U}(s) \quad (5.11)$$

und schließlich ist die Übertragungsfunktion der Modellfolgeregelung

$$\mathbf{X}(s) = \mathbf{G}_3 \mathbf{U}(s). \quad (5.12)$$

Zusammen mit dem Regler  $\mathbf{G}_{K_3}$  und der Übertragungsfunktion der Umgebung  $\mathbf{G}_E$  ergibt sich die postulierte Übertragungsfunktion der Kraft-Momentenregelung aus Gleichung (5.7).

### 5.1.2 Kraft-Momentenvorsteuerung

Im Umgebungskontakt treten externe Kräfte auf, die am Eingang der Regelstrecke nach den Motorumrichtern eingreifen und die Struktur des Regelkreises verändern. Um diesem Störeingriff entgegen zu wirken und weiterhin die Gültigkeit der mit (5.7) beschriebenen Systemdynamik gewährleisten zu können, wird eine Kraft-Momentenvorsteuerung eingeführt.

Wie in Abbildung 5.1 zu sehen, greift diese möglichst nahe an den Störgrößen an und kompensiert, bei geeigneter Wahl der Vorsteuergrößen, deren Einfluss.

**Korollar 4 (Kraft-Momentenvorsteuerung)** *Befindet sich der Parallelroboter im Umgebungskontakt, garantiert die Vorsteuerung der erwarteten Umgebungskräfte und -momente*

$$\mathbf{f}_{\text{fwd}}(t) := \mathbf{f}(t)$$

*für den Fall, dass die Bandbreite der Umgebungskräfte wesentlich kleiner als die Umrichter verzögerung ist, ein Verhalten der unterlagerten Antriebsregelung wie im Freiraum mit*

$$\mathbf{X}(s) = \mathbf{G}_3 \mathbf{U}(s),$$

*da die auftretenden Umgebungskräfte kompensiert werden.*

**BEWEIS (KRAFT-MOMENTENVORSTEUERUNG)** Zum Beweis der Kompensation durch die Kraft-Momentenvorsteuerung ist es zweckmäßig die Gesamtübertragungsfunktion  $\mathbf{G}$  in die Verzögerung der Umrichter und die Bewegungsgleichung aufzuteilen

$$\mathbf{G} = \mathbf{G}' \cdot \left\{ \frac{1}{T_c s + 1} \right\}_n.$$

Dabei ist  $\mathbf{G}'$  die Beschreibung der Mechanik und  $\left\{ \frac{1}{T_c s + 1} \right\}_n$  die der Umrichter in allen  $n$  Freiheitsgraden. Die Übertragungsfunktion vom Eingang des Führungsgrößengenerators zum Ausgang des Parallelroboters lautet

$$\mathbf{X}(s) = \mathbf{G}_3 \mathbf{U}(s) + \mathbf{G}' \left\{ \frac{1}{T_c s + 1} \right\}_n (\mathbf{I} + \mathbf{G} \mathbf{G}_{K_{1,2}})^{-1} \mathbf{F}_{\text{fwd}}(s) - \mathbf{G}' (\mathbf{I} + \mathbf{G} \mathbf{G}_{K_{1,2}})^{-1} \mathbf{F}(s).$$

Ist die Bandbreite der Umgebungskräfte wesentlich kleiner als die Zeitkonstante  $T_c$ , kann die Umrichter verzögerung vernachlässigt werden und die beiden letzten Terme heben sich auf. Die Übertragungsfunktion

$$\mathbf{X}(s) = \mathbf{G}_3 \mathbf{U}(s)$$

entspricht der des Parallelroboters im Freiraum.

Die Kompensation der externen Kräfte ist durch das maximale Drehmoment der Antriebe begrenzt und reicht, je nach Umgebungssteifigkeit von einigen Mikrometern bis zu wenigen Millimetern Eindringtiefe. In der Literatur, so z. B. in Sciavicco und Siciliano [2000], kompensieren häufig die gemessenen Kräfte das veränderte Systemverhalten im Umgebungskontakt. Dadurch ist eine exakte Linearisierung der Strecke zu jedem Zeitpunkt der Kontaktphase und nicht nur im stationären Fall gegeben. Allerdings führt diese Strategie auf eine Regelkreisstruktur mit positiver Rückkopplung der externen Kräfte und damit zu potentiellen



Instabilitäten. Wie in dem Aufsatz von McClamroch und Wang [1988, S. 421 unten] ausführlich dargelegt, wurde im Rahmen dieser Arbeit darauf verzichtet.

### 5.1.3 Strukturumschaltende Regelung

In dem Kapitel 4 über die Systemdynamik im Umgebungskontakt wurden die Bewegungsgleichungen für jede der drei in Abbildung 1.1 dargestellten Phasen der Kontaktregelung hergeleitet. Die Bewegungsgleichungen unterscheiden sich in ihrer Struktur, so dass jeder Phase ein eigenes Regelungsgesetz zugeordnet wird, das die spezifischen Aufgaben erfüllt [vgl. Assuncao und Schumacher, 2003]. Das Bewegungsmodul der Kontaktregelung beinhaltet einen endlichen Automaten, der die Umschaltung zwischen den Reglern realisiert.

In der Freiraumbewegung ist ein Positionsplaner als Bewegungsmodul im Eingriff [Kolbus u. a., 2005]. Ein Aktionsprimitiv bringt den Endeffektor möglichst nahe an die Umgebung heran, ohne dass diese berührt wird.

Das folgende Aktionsprimitiv ruft das Bewegungsmodul der Kontaktregelung auf. Die Transition von der Freiraumbewegung in den sicheren Umgebungskontakt kann in die zwei Intervalle der Annäherung des Endeffektors an die Umgebung und die Dämpfung der Abpralleffekte unterteilt werden. Zum Zeitpunkt  $t_0$  bis zum ersten Umgebungskontakt  $t_k$  befindet sich der Endeffektor im Freiraum und fährt geschwindigkeitsgeregelt an die Oberfläche heran. Die Detektion des Umgebungskontakts durch die Messung der Endeffektorkräfte modifiziert die Solltrajektorie so, dass die unterlagerte Antriebsregelung die Abpralleffekte dämpfen kann und der Endeffektor zum Zeitpunkt  $t_d$  sicher im Umgebungskontakt verharret.

Im Umgebungskontakt ist die Kraft-Momentenregelung im Eingriff, die die gewünschte Endeffektorkraft einstellt und in dieser sicheren Position verharret, bis die Abbruchbedingungen des Aktionsprimitivs erfüllt sind. Verliert der Endeffektor während der Regelung den Kontakt mit der Umgebung beginnt der Zyklus in der Transitionsphase von Neuem. Dies verhindert *Wind-up*-Effekte der Kraft-Momentenregelung. Abweichend von dem oben beschriebenen idealtypischen Verhalten kann sich der Endeffektor zu Beginn der Kontaktregelung schon im Umgebungskontakt befinden. In diesem Fall wird sofort zur Kraft-Momentenregelung umgeschaltet.

Die Abbildung 5.4 zeigt den endlichen Automaten der Kontaktregelung für einen Freiheitsgrad. Die Definition des Eingabealphabets entspricht der eines idealen Reglers mit positivem Kraftsollwert. In allgemeiner Form ist die strukturumschaltende Regelung wie folgt definiert:

**Definition 2 (Endlicher Automat der Kontaktregelung)** Der endliche Automat der strukturumschaltenden Kontaktregelung

$$\text{EA}^{(k)} := \{\mathcal{Z}, \mathcal{Z}_0, \mathcal{E}, \mathcal{U}, \mathcal{A}\} \quad (5.13)$$

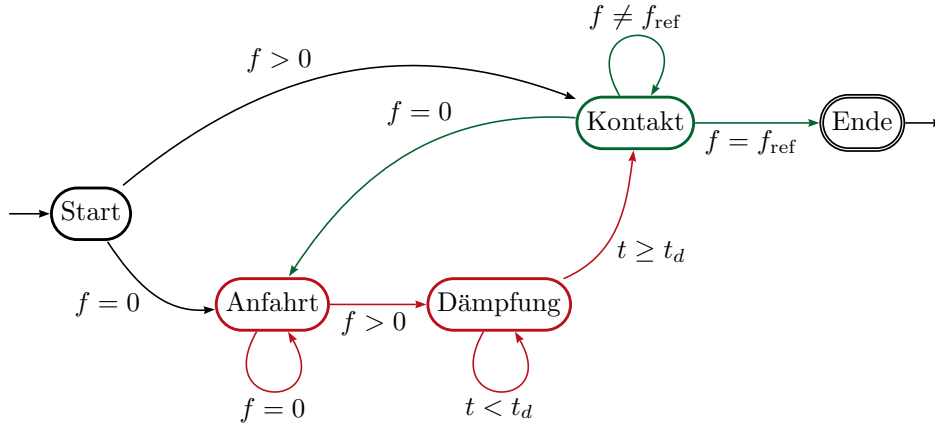


Abb. 5.4: Endlicher Automat des Kontaktreglers mit vereinfachtem Eingabealphabet

für einen Freiheitsgrad  $k$  des Task-Frames besitzt die Zustände

$$\mathcal{Z} := \{\text{Start}, \text{Anfahrt}, \text{Dämpfung}, \text{Kontakt}, \text{Ende}\},$$

wobei  $\mathcal{Z}_0 = \text{Start}$  den Anfangszustand darstellt. Das Eingabealphabet  $\mathcal{E}$  ist in Tabelle 5.1 definiert. Dabei wird die  $\varepsilon_1$ -Umgebung so festgelegt, dass das Rauschen auf den Signalen die Funktion nicht stört. Die  $\varepsilon_2$ -Umgebung legt fest, wann das Regelziel erreicht ist. Das Ende der Dämpfungsphase und den Übergang zur Kraft-Momentenregelung markiert die Zeit  $t_d$ .

Eingang	Definition	Beschreibung
$g_1$	$ f  \geq \varepsilon_1 \vee  f_{\text{ref}}  \leq \varepsilon_1$	Endeffektor im Umgebungskontakt
$\neg g_1$	$ f  < \varepsilon_1 \wedge  f_{\text{ref}}  > \varepsilon_1$	Endeffektor im Freiraum
$g_2$	$t \geq t_d$	Zeit erreicht
$\neg g_2$	$t < t_d$	Zeit nicht erreicht
$g_3$	$ f_{\text{ref}}  - \varepsilon_2 \leq  f  \leq  f_{\text{ref}}  + \varepsilon_2$	Regelziel erreicht
$\neg g_3$	$ f_{\text{ref}}  - \varepsilon_2 >  f  >  f_{\text{ref}}  + \varepsilon_2$	Regelziel nicht erreicht

Tab. 5.1: Eingabealphabet  $\mathcal{E}$  des endlichen Automaten

Die Zustandsübergangsfunktion  $\mathcal{U}$  in Tabelle 5.2 definiert den Folgezustand, den der Automat aufgrund einer bestimmten Eingabe annimmt.

Eingang	Start	Anfangszustand		
		Anfahrt	Dämpfung	Kontakt
$g_1$	Kontakt	Dämpfung		
$\neg g_1$	Anfahrt	Anfahrt		Anfahrt
$g_2$			Kontakt	
$\neg g_2$			Dämpfung	
$g_3$				Ende
$\neg g_3$				Kontakt

Tab. 5.2: Zustandsübergangsfunktion  $\mathcal{U}$  des endlichen Automaten

Jedem Zustand ist durch das Ausgabealphabet  $\mathcal{A}$  in Tabelle 5.3 ein Ausgang zugeordnet. Dieser entspricht der Solltrajektorie eines Bewegungsmoduls in einem Freiheitsgrad.

Zustand	Ausgang	Beschreibung
Start	$T_0$	sichere Trajektorie des vorhergehenden Aktionsprimitivs
Anfahrt	$T_{\text{ref},a}$	Trajektorie mit konstanter Geschwindigkeit
Dämpfung	$T_{\text{ref},d}$	Ruhelage während der Dämpfungsphase
Kontakt	$T_{\text{ref},k}$	Stelltrajektorie des Kraft-Momentenreglers
Ende	$T_0$	sichere Trajektorie des Kontaktreglers

Tab. 5.3: Ausgabealphabet  $\mathcal{A}$  des endlichen Automaten

Die Erzeugung der definierten Trajektorien ist Aufgabe der Regelungsgesetze, die in den folgenden Abschnitten beschrieben werden. Entgegen der zeitlichen und logischen Abfolge wird dabei zuerst das Regelungsgesetz für den Kontaktfall erläutert, das es eine umfassendere Betrachtung und Herleitung des Führungsgrößengenerators beinhaltet. Die Regelung der Transitionsphase nutzt einen vereinfachten Führungsgrößengenerator und wird anschließend erläutert.

## 5.2 Kraft-Momentenregler

Der Kraft-Momentenregler ist der wesentliche Teil der Kontaktregelung. Er besteht aus dem in Abbildung 5.1 dargestellten Führungsgrößengenerator, dem eigentlichen Regelungsgesetz und der Kraft-Momentenvorsteuerung.

### 5.2.1 Führungsgrößengenerator

Der Führungsgrößengenerator erzeugt aus der vom Kraft-Momentenregler vorgegebenen Stellgröße eine ruckbegrenzte Solltrajektorie im Task-Frame des aktuellen Aktionsprimitivs. Durch den Task-Frame-Formalismus der Steuerungsarchitektur wird diese Solltrajektorie in das Roboterkoordinatensystem transformiert und das inverse dynamische Modell der

Vorsteuerung generiert daraus die Vorsteuergrößen der unterlagerten Antriebsregelung. Die Ruckbegrenzung der Solltrajektorie lässt sich durch die Anwendung eines Systems dritter Ordnung erreichen. Sie ist notwendig, um eine Zerstörung der Parallelroboterstruktur bei hochdynamischen Bewegungen zu verhindern. Das zur Anwendung als Führungsgrößengenerator vorgesehene dynamische System muss das Verhalten der unterlagerten Antriebsregelung, wie in Gleichung (5.5) gefordert, möglichst genau nachbilden, stationär genau sein und darf keine Nullstellen aufweisen, damit dessen Ausgang und die Zustandsgrößen gleichzeitig die Elemente der Solltrajektorie beschreiben. Für die Kontaktregelung wird der Führungsgrößengenerator daher wie folgt definiert:

**Definition 3 (Führungsgrößengenerator)** Die unterlagerte Antriebsregelung wird für den  $k$ -ten Freiheitsgrad als System dritter Ordnung

$$G_3^{(k)}(s) = \frac{X_{\text{ref}}^{(k)}(s)}{U^{(k)}(s)} = \frac{1}{(T_a s + 1) \left( \frac{s^2}{\omega_{0,a}^2} + \frac{2D_a}{\omega_{0,a}} s + 1 \right)} \quad (5.14)$$

mit einer reellen Polstelle und einem konjugiert komplexen Polpaar approximiert. Unter Berücksichtigung des multiplikativen Unsicherheitsfaktors des Massenträgheitsmoments aus Gleichung (3.61) und der Führungsübertragungsfunktion (3.62) existieren die Parameter

$$T_a, \omega_{0,a} \text{ und } D_a \in \mathbb{R}_+,$$

so dass gilt:

- (a) Der Ausgang  $x_{\text{ref}}^{(k)}$  stellt für alle  $\Delta^{(k)}$  eine stabile und ausreichend langsame Annäherung der Führungsübertragungsfunktion (3.62) dar.
- (b) Der Ausgang bzw. die Zustandsgrößen definieren eine ruckbegrenzte Solltrajektorie

$$T_{\text{ref}}^{(k)} = \left\{ x_{\text{ref}}^{(k)}, \dot{x}_{\text{ref}}^{(k)}, \ddot{x}_{\text{ref}}^{(k)} \right\}, \quad k \in [1; n]_{\mathbb{N}}$$

des Führungsgrößengenerators in kartesischen Koordinaten.

Ausgangspunkt der Auslegung des Führungsgrößengenerators ist die bekannte Führungsübertragungsfunktion der unterlagerten Antriebsregelung

$$F_{gw}^{(k)}(s) = \frac{Q^{(k)}(s)}{Q_{\text{ref}}^{(k)}(s)} = \frac{(T_m s + 1)(4T_c s + 1)}{(T_c s + 1) \left[ 256\Delta^{(k)} T_m T_c^4 s^5 + 256\Delta^{(k)} T_c^3 (1 + T_m T_c) s^4 \right.} \dots$$

$$\dots \left. \frac{\dots}{+ 256\Delta^{(k)} T_c^3 s^3 + 4T_c(24T_c + T_m) s^2 + (16T_c + T_m) s + 1} \right]. \quad (3.62)$$

Im Gegensatz zu der in Gleichung (5.14) definierten Übertragungsfunktion im kartesischen Task-Frame beschreibt sie ein Übertragungsverhalten im Roboterkoordinatensystem. Mittels des direkten bzw. inversen kinematischen Problems können beide Darstellungen ineinander überführt werden. Die Transformationen enthalten jedoch keine weiteren dynamischen Systeme, so dass zwar die stationären Endwerte in den unterschiedlichen Koordinatensystemen nicht übereinstimmen, wohl aber die, für die Beschreibung des Führungsgrößengenerators einzig notwendigen, dynamischen Parameter.

Die Verzögerung der Umrichter  $T_c$  und Messeinrichtung  $T_m$  in der Führungsübertragungsfunktion (3.62) stellen für jedes System Konstanten dar, die für alle Antriebe und damit auch für alle Variablen des Roboterkoordinatensystems gleich sind. Das Einschwingverhalten, und damit die dynamischen Parameter, ist eine Funktion des Unsicherheitsfaktors  $\Delta^{(k)}$ . In Abbildung 5.5(a) ist die Wurzelortskurve einer typischen Implementierung der unterlagerten Antriebsregelung mit der charakteristischen Gleichung

$$1 + \Delta^{(k)} \cdot F_{gw}^{(k)'} = 1 + \Delta^{(k)} \cdot \frac{256(T_c s + 1)(T_m s + 1)T_c^3 s^3}{(4T_m + 96T_c)T_c s^2 + (16T_c + T_m)s + 1} \quad (5.15)$$

dargestellt. Das angewandte Verfahren zur Ordnungsreduktion der Führungsübertragungsfunktion der unterlagerten Antriebsregelung ermittelt die drei dominierenden Polstellen anhand der Wurzelortskurve. In der Diskussion sind drei Fälle zu unterscheiden.

Fall 1: Für ein System mit  $\Delta^{(k)} \in [\Delta_{\min}^{(k)}; \Delta_{\text{krit}}^{(k)})$  existieren zwei reelle Polstellen und zwei konjugiert komplexe Polpaare. Das Polpaar mit der geringeren Eigenfrequenz dominiert das Systemverhalten.

Fall 2: Bei  $\Delta_{\text{krit}}^{(k)}$  vereinigt sich ein konjugiert komplexes Polpaar zu einer doppelten reellen Polstelle. Außerdem sind zwei weitere reelle Polstellen und ein konjugiert komplexes Polpaare vorhanden. Das Systemverhalten wird von der doppelten Polstelle dominiert.

Fall 3: Ein Unsicherheitsfaktor  $\Delta^{(k)} \in (\Delta_{\text{krit}}^{(k)}; \Delta_{\max}^{(k)}]$  führt auf vier reelle Nullstellen und ein konjugiert komplexes Polpaar. In diesem Fall dominiert das einzige noch vorhandene Polpaar und die Polstelle mit der größten Zeitkonstanten das Systemverhalten.

Die Auslegung des Führungsgrößengenerators erfolgt mit den bei  $\Delta_{\text{krit}}^{(k)}$  dominierenden Polen (vgl. Fall 2): Die doppelte Nullstelle wird zu der Verzögerung erster Ordnung mit der Zeitkonstante  $T_a$  zusammengefasst und die konjugiert komplexen Polstellen werden in das angenäherte Modell übernommen. Dieses Verfahren zu Ordnungsreduktion führt auf ein stabiles und für alle Unsicherheiten  $\Delta^{(k)} \in [\Delta_{\min}^{(k)}; \Delta_{\max}^{(k)}]$  ausreichend langsames Verhalten, da

(a) die angenäherte Zeitkonstante größer als die maximal auftretende Zeitkonstante ist und

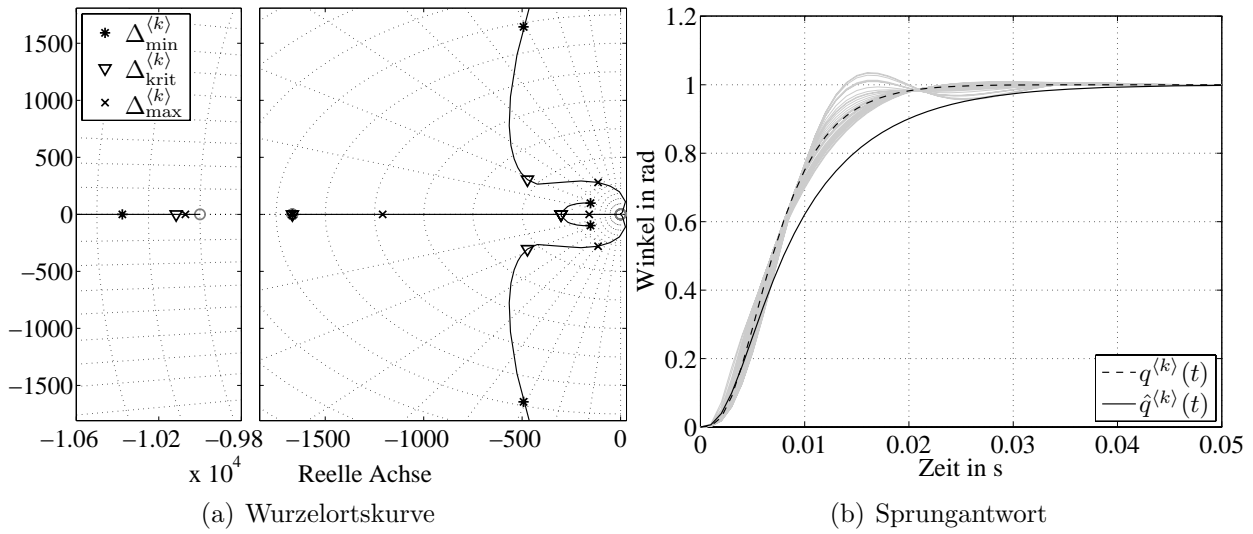


Abb. 5.5: Verfahren zur Ordnungsreduktion der Führungsübertragungsfunktion der unterlagerten Antriebsregelung und zur Auslegung des Führungsgrößengenerators

(b) die konjugiert komplexen Pole auf ihren Zweigen der Wurzelortskurve für  $\Delta_{\text{krit}}^{(k)}$  die größte Dämpfung aufweisen.

Die Sprungantworten in Abbildung 5.5(b) bestätigen diese These.

**Beispiel 3 (Parameter des Führungsgrößengenerators)** Eine Implementierung der unterlagerten Antriebsregelung hat eine Umrichterzeitkonstante von  $T_c = 600 \mu\text{s}$  und eine Messverzögerung von  $T_m = 100 \mu\text{s}$ . Die Modellunsicherheit wird mit  $\pm 70\%$  angegeben, so dass  $\Delta^{(k)} \in [0,3; 1,7]$  ist.

Die Analyse des Systemverhaltens mittels der charakteristischen Gleichung (5.15) und der Wurzelortskurve ergeben einen dominierenden Pol bei  $T_a = 7,49 \text{ ms}$  und ein konjugiert komplexes Polpaar mit  $\omega_{0,a} = 625,23 \text{ rad/s}$  und  $D_a = 0,81$ .

Der kritische Unsicherheitsfaktor ist in dieser Konfiguration  $\Delta_{\text{krit}}^{(k)} = 1,029$ .

Eine Begrenzung der Zustandsgrößen des Führungsgrößengenerators ist nicht notwendig, da diese die Lage des Parallelroboters im Arbeitsraum abbilden und von den Sicherheitsfunktionen der Steuerungsarchitektur überwacht werden. Ein zu großer Schleppfehler, d. h. eine Diskrepanz zwischen Ist- und Sollwerten sowie das Verlassen des Arbeitsraums führen zur Notbremsung des Parallelroboters. Eine Initialisierung der Zustandsgrößen des Führungsgrößengenerators erfolgt mit jedem neuen Aktionsprimitiv.

### 5.2.2 Kraft-Momentenregler mit statischem Umgebungsmodell

Der Kraft-Momentenregler berechnet aus der Sollwertvorgabe des Aktionsprimitivs und der gemessenen Endeffektorkraft im Task-Frame die Stellgröße, aus der der Führungsgrößengenerator die Solltrajektorie  $T_{\text{ref},k}$  generiert. Befindet sich der Endeffektor des Parallelroboters

im Kontakt mit einer Umgebung, die durch ein statisches Modell beschrieben werden kann, wird der Regler wie folgt definiert:

**Definition 4 (Kraft-Momentenregler I)** Mit dem in Gleichung (5.14) als PT<sub>3</sub>-System definierten Führungsgrößengenerator wird der Kraft-Momentenregler als PI-Regler mit der Übertragungsfunktion

$$G_{K_3}^{(k)}(s) := V_3^{(k)} \frac{T_{i,3}^{(k)} s + 1}{T_{i,3}^{(k)}}, \quad k \in [1; n]_{\mathbb{N}} \quad (5.16)$$

ausgeführt. Dabei sind der Verstärkungsfaktor  $V_3^{(k)}$  und die Integrationszeitkonstante  $T_{i,3}^{(k)}$  die Parameter, die während der Auslegung angepasst werden.

Ausgangspunkt der Reglerauslegung ist das Kraft-Momenten-Regelungskonzept aus Kollar 3 und die Beschreibung des statischen Umgebungsmodells in Abschnitt 4.3.1. Dementsprechend ergibt sich die Übertragungsfunktion des offenen Kreises für eine Achse  $k$  des Task-Frames zu

$$F_{o,3}^{(k)} = G_{K_3}^{(k)}(s) G_3^{(k)}(s) G_E^{(k)}(s) = V_3^{(k)} \frac{T_{i,3}^{(k)} s + 1}{T_{i,3}^{(k)}} \frac{c_e^{(k)} l^{(k)^2}}{(T_a s + 1) \left( \frac{s^2}{\omega_{0,a}^2} + \frac{2D_a}{\omega_{0,a}} s + 1 \right)}. \quad (5.17)$$

Das Kürzen der Nullstelle des PI-Reglers mit  $T_{i,3} := T_a$  führt auf

$$F_{o,3}^{(k)} = \frac{V_3^{(k)} \cdot c_e^{(k)} l^{(k)^2}}{T_a s \left( \frac{s^2}{\omega_{0,a}^2} + \frac{2D_a}{\omega_{0,a}} s + 1 \right)}. \quad (5.18)$$

Es bleibt damit die Reglerverstärkung  $V_3^{(k)}$  als einziger freier Parameter, die durch die Vorgabe eines gewünschten Phasenrands [vgl. Föllinger, 1990] berechnet wird. Für die Momentenregelung der rotatorischen Freiheitsgrade muss zusätzlich die Wirkung des Hebelarms berücksichtigt werden, so dass dieser Parameter in den Gleichungen zur Reglerauslegung immer mitgeführt wird. Im Falle der Kraftregelung wird  $l$  gleich eins gesetzt und hat somit keine Wirkung.

Der Phasenrand des offenen Regelkreises (5.18) berechnet sich zu

$$\varphi_m = \angle G(j\omega) + \pi = \arctan \left( \frac{\omega_{0,a}^2 - \omega^2}{2D_a \omega \omega_{0,a}} \right). \quad (5.19)$$

Gemeinsam mit der Amplitudenbedingung

$$|G(j\omega)| = \frac{V_3^{(k)} c_e^{(k)} l^{(k)^2}}{T_a \omega \sqrt{\left(1 - \frac{\omega^2}{\omega_{0,a}^2}\right)^2 + \frac{4D_a \omega^2}{\omega_{0,a}^2}}} = 1 \quad (5.20)$$

ergibt sich ein Gleichungssystem zur Bestimmung der Reglerverstärkung dessen gültige Lösung

$$V_3^{(k)} = \frac{2D_a T_a \omega_{0,a}}{c_e^{(k)} l^{(k)^2}} \sqrt{1 + \tan(\varphi_m)^2} \left( \sqrt{1 + D_a^2 \tan(\varphi_m)^2} - D_a \tan(\varphi_m) \right) \dots \\ \dots \sqrt{1 + 2D_a^2 \tan(\varphi_m)^2 + 2D_a \tan(\varphi_m) \sqrt{1 + D_a^2 \tan(\varphi_m)^2}} \quad (5.21)$$

ist und so den Verstärkungsfaktor bestimmt. Dieser ist abhängig von den Parametern des Führungsgrößengenerators, der Umgebung und dem Hebelarm  $l^{(k)}$ .

Das Vorsteuersignal der Kraft-Momentenregelung wird verzögert, um es dem Einschwingvorgang des Regelkreises anzupassen und somit möglichst wenige Störeinflüsse zu generieren. Dazu wird die Übertragungsfunktion des geschlossenen Kreises

$$F_{g,3}^{(k)} = \frac{V_3^{(k)} \omega_{0,a}^2 c_e^{(k)} l^{(k)^2}}{T_a s^3 + 2T_a D_a \omega_{0,a} s^2 + T_a \omega_{0,a}^2 s + V_3^{(k)} \omega_{0,a}^2 c_e^{(k)} l^{(k)^2}} \quad (5.22)$$

durch ein PT<sub>1</sub>-System

$$G_{\text{fwd},3} = \frac{1}{T_{\text{fwd},3} s + 1} \quad (5.23)$$

mit der Ersatzzeitkonstante

$$T_{\text{fwd},3} = \frac{T_a}{V_3^{(k)} c_e^{(k)} l^{(k)^2}} \quad (5.24)$$

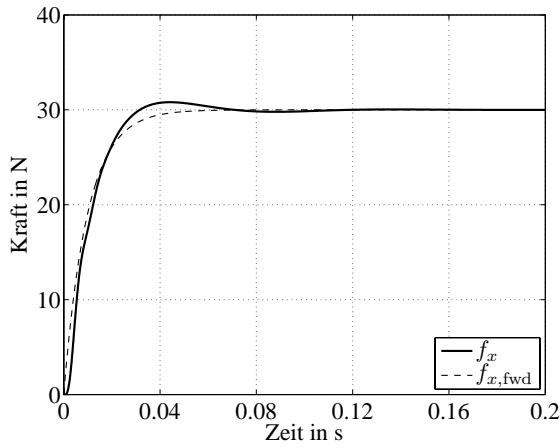
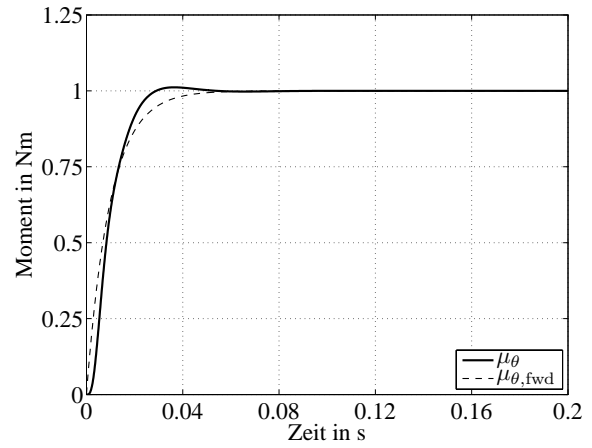
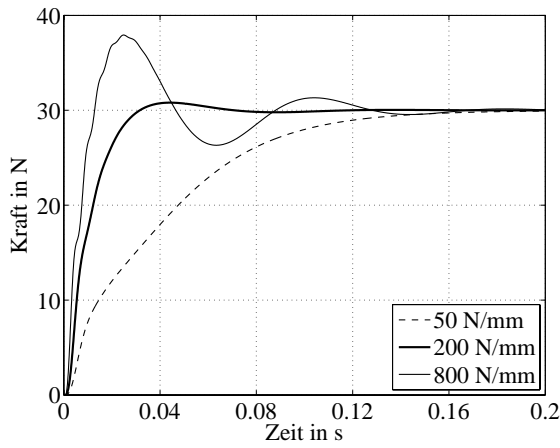
angenähert. Der Verstärkungsfaktor ist umgekehrt proportional zum Quadrat des Hebelarms  $V^{(k)} \propto \frac{1}{l^{(k)^2}}$ . Dadurch ist die Verzögerung unabhängig von dem wirksamen Hebelarm  $l$  und somit gleich für die Kraft- und Momentenregelung.

**Beispiel 4 (Auslegung des PI-Kraft-Momentenreglers)** Gegeben ist eine Umgebungssteifigkeit von  $c_{e,0} = 200 \text{ N/mm}$ , ein wirksamer Hebelarm von  $l^{(\varphi)} = 50 \text{ mm}$  und die in Beispiel 3 ermittelten Parameter des Führungsgrößengenerators.

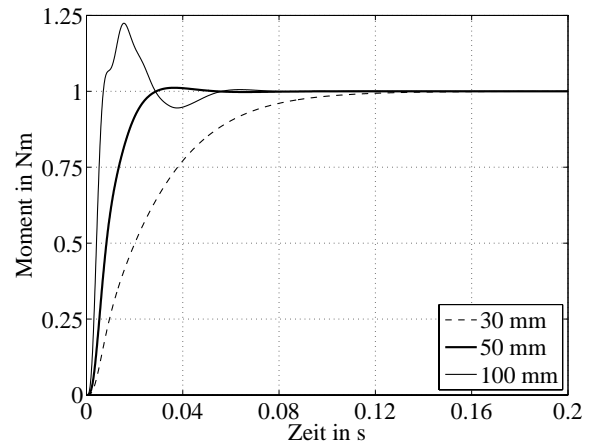
Die Verstärkung beträgt, bei einem gewünschten Phasenrand von  $75^\circ$ , für die translatorischen Freiheitsgrade  $V_3^{(x)} = 3,8 \cdot 10^{-6}$  und für rotatorische Freiheitsgrade  $V_3^{(\varphi)} = 1,52 \cdot 10^{-3}$ . Die Verzögerung der Vorsteuergröße ist  $T_{\text{fwd},3} = 7,49 \text{ ms}$ .

Die Funktion des entwickelten Kraft-Momentenregelungskonzepts und der Einfluss der unterschiedlichen Strecken- und Entwurfparameter auf das Verhalten der Regelung wurde in Simulationen untersucht, deren Ergebnisse in Abbildung 5.6 dargestellt sind. Die Simulationen wurden mit dem dynamischen Modell des in Anhang A.1 beschriebenen Parallelroboters HEXA durchgeführt. In Abbildung 5.6(a) und (b) ist die Sprungantwort der Kraftregelung in der  $x$ -Achse bzw. eine Momentenregelung um die  $y$ -Achse gezeigt. In beiden Fällen be-

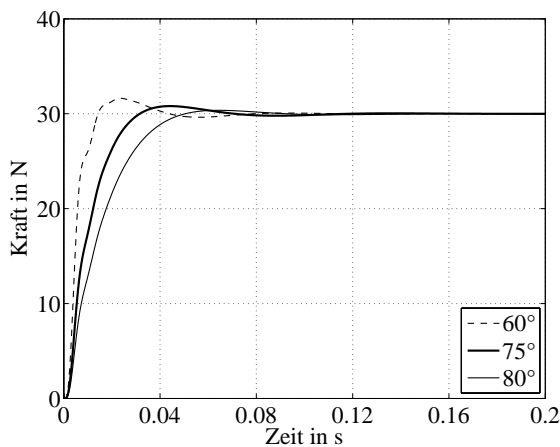
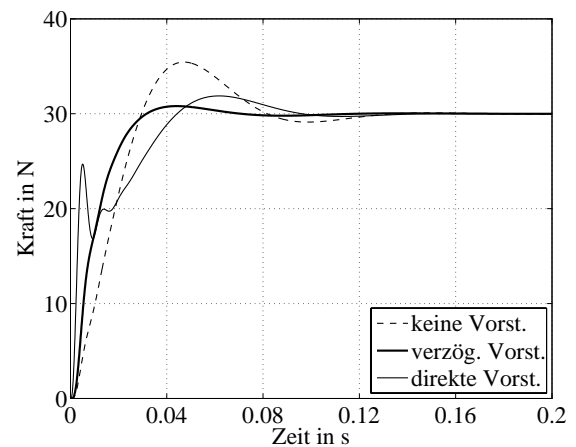


(a) Sprungantwort der Kraftregelung in der  $x$ -Achse(b) Sprungantwort der Momentenregelung um die  $y$ -Achse

(c) Sprungantwort der Kraftregelung für verschiedene Umgebungssteifigkeiten



(d) Sprungantwort der Momentenregelung für verschiedene Hebelarme

(e) Auswirkungen des Entwurfsparameters *Phasenrand* auf die Sprungantwort der Kraftregelung

(f) Auswirkung der Vorsteuerung auf die Sprungantwort der Kraftregelung

Abb. 5.6: Simulationsergebnisse der Kraft-Momentenregelung mit statischem Umgebungsmodell

trägt die Umgebungssteifigkeit  $c_{e,0} = 200 \text{ N/mm}$ . Nach 100 ms ist der Kraftsollwert von  $f_{x,\text{ref}} = 30 \text{ N}$  in Abbildung 5.6(a) erreicht. Der Verlauf folgt der Vorsteuerung  $f_{x,\text{fwd}}$  mit leichten Überschwingungen. Gleiches gilt für die Momentenregelung mit einem Hebelarm von  $l_x = 50 \text{ mm}$  und dem Sollwert von  $\mu_\theta = 1 \text{ Nm}$  in Abbildung 5.6(b).

Die Sprungantworten für verschiedene Umgebungssteifigkeiten in Abbildung 5.6(c) geben Auskunft über die Robustheit der Kraft-Momentenregelung. Ein für die Umgebungssteifigkeit von  $c_{e,0} = 200 \text{ N/mm}$  ausgelegter Regler führt bei einer weicheren Umgebung zu einem Verlust der Dynamik, während eine steifere Umgebung größere Überschwingungen verursacht. Die Simulationen zeigen jedoch, dass der Einschwingvorgang innerhalb von 200 ms abgeschlossen ist, wenn sich die Umgebungssteifigkeit in einem Bereich von  $c_{e,0} \in [50 \text{ N/mm}; 800 \text{ N/mm}]$  bewegt. Für die Momentenregelung ist die Länge des Hebelarms  $l$  ein kritischer Parameter, der während einer Montageaufgabe nicht immer bekannt ist. Abbildung 5.6(d) zeigt die Sprungantworten der für  $l_x = 50 \text{ mm}$  ausgelegten Momentenregelung mit unterschiedlichen tatsächlichen Hebelarmen. Ein kürzerer Hebelarm hat eine langsamere Momentenregelung zur Folge. Demgegenüber verursacht ein längerer Hebelarm ein deutliches Überschwingen. Die Momentenregelung ist jedoch in allen Fällen stabil und erreicht spätestens nach 150 ms den Endwert von  $\mu_\theta = 1 \text{ Nm}$ .

Der Entwurfparameter des Phasenrands hat ebenfalls einen Einfluss auf den Verlauf der Regelgröße. Abbildung 5.6(e) zeigt die Sprungantworten der Kraftregelung mit einer Umgebungssteifigkeit von  $c_{e,0} = 200 \text{ N/mm}$ . Je geringer der Phasenrand, desto deutlicher tritt das Überschwingen auf. Die Einschwingzeit wird dagegen kaum beeinflusst und liegt für alle drei Sprungantworten bei ca. 100 ms. Der empirisch ermittelte optimale Wert des Phasenrandes liegt bei  $\varphi_m \approx 75^\circ$ .

Abschließend soll die Auswirkung der Kraft-Momentenvorsteuerung betrachtet werden. Die Ersatzzeitkonstante  $T_{\text{fwd},3}$  ist unabhängig von der Umgebungssteifigkeit und dem Hebelarm und beträgt für die in Beispiel 4 berechnete Reglerauslegung 7,49 ms. Die Umrichterzeitkonstante wurde zu  $T_c = 100 \mu\text{s}$  angenommen, so dass die Bedingungen in Korollar 4 erfüllt sind. Wird die Kraftregelung ohne eine Vorsteuerung betrieben, führt das zu einem deutlichen Überschwingen mit der Amplitude von  $f_{x,\text{max}} = 35 \text{ N}$ , da die unterlagerte Antriebsregelung die Umgebungskräfte kompensieren muss. Eine direkte Aufschaltung des Kraftsollwerts verursacht einen raschen Anstieg der Regelgröße zu Beginn und einer geringfügig größeren Überschwingung. Durch die zusätzlich angeregten transienten Vorgänge dauert der Einschwingvorgang länger.

Der vorgestellte Kraft-Momentenregler nach Definition 4 stellt eine robuste Realisierung des entwickelten Regelungskonzepts dar. Neben den Simulationen erlauben die durchgeführten experimentellen Untersuchungen eine weitere Beurteilung. Deren Diskussion erfolgt in Kapitel 6.

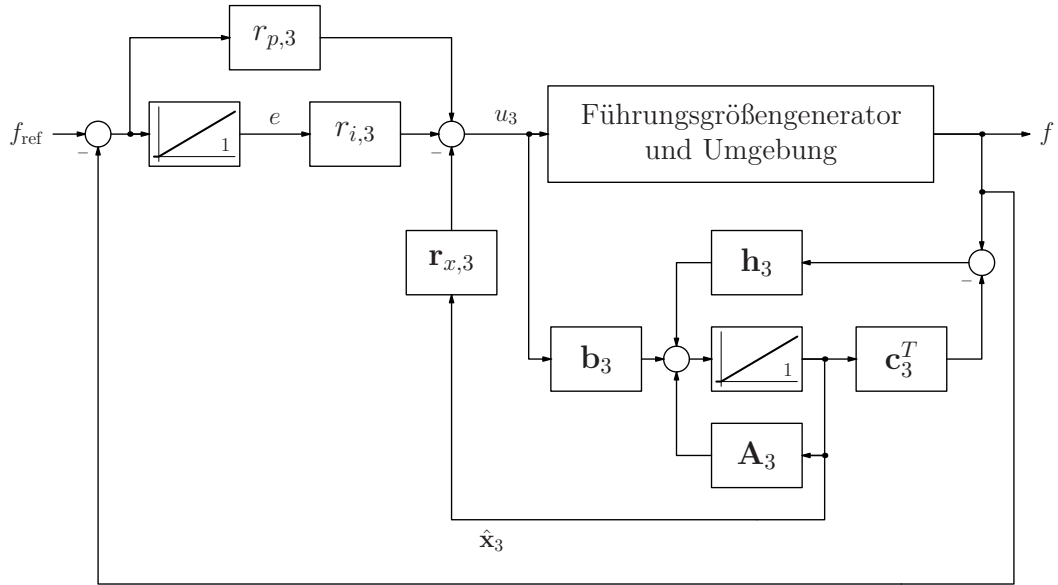


Abb. 5.7: Dynamischer Zustandsregler mit Beobachter für die Regelstrecke bestehend aus Führungsgrößengenerator und dynamischem Umgebungsmodell

### 5.2.3 Kraft-Momentenregler mit dynamischem Umgebungsmodell

Das dynamische Umgebungsmodell ermöglicht eine Beschreibung des Systemverhaltens mit einer zusätzlichen Nachgiebigkeit, die sowohl der Umgebung als auch dem Parallelroboter selbst zugeordnet werden kann. Für diese komplexe Regelstrecke wird der Kraft-Momentenregler wie folgt definiert:

**Definition 5 (Kraft-Momentenregler II)** Für den in Gleichung (5.14) als  $PT_3$ -System definierten Führungsgrößengenerator und das in Abschnitt 4.3.1 beschriebene dynamische Umgebungsmodell wird der Kraft-Momentenregler als *dynamischer Zustandsregler mit Beobachter* ausgeführt.

Das Blockschaltbild des vollständigen Systems für einen Freiheitsgrad, dessen Elemente in den folgenden Abschnitten beschrieben werden, ist in Abbildung 5.7 dargestellt.

#### Regelstrecke

Die Regelstrecke besteht aus dem Führungsgrößengenerator und dem dynamischen Umgebungsmodell. Deren Übertragungsfunktionen für einen Freiheitsgrad  $k$  des Parallelroboters lauten

$$G_3^{(k)}(s) = \frac{1}{(T_a s + 1) \left( \frac{s^2}{\omega_{0,a}^2} + \frac{2D_a}{\omega_{0,a}} s + 1 \right)} \quad (5.14)$$

und

$$G_E^{(k)}(s) = \frac{c_{e,0}(m_{e,1}s^2 + d_{e,1}s + c_{e,1})}{m_{e,1}s^2 + d_{e,1}s + c_{e,0} + c_{e,1}} \quad \text{mit } k \in [1; n]_{\mathbb{N}}. \quad (4.38)$$

Die Reihenschaltung beider Übertragungsfunktionen in Normalform ist schließlich

$$G_3^{(k)}(s)G_E^{(k)}(s) = \frac{b_2s^2 + b_1s + b_0}{s^5 + a_4s^4 + a_3s^3 + a_2s^2 + a_1s + a_0}, \quad (5.25)$$

wobei die Koeffizienten mit

$$\begin{aligned} a_0 &= \frac{\omega_{0,a}^2(c_{e,0} + c_{e,1})}{T_a m_{e,1}}, \\ a_1 &= \frac{\omega_{0,a}^2 d_{e,1} + (2D_a \omega_{0,a} + T_a \omega_{0,a}^2)(c_{e,0} + c_{e,1})}{T_a m_{e,1}}, \\ a_2 &= \frac{\omega_{0,a}^2 m_{e,1} + (2D_a \omega_{0,a} + T_a \omega_{0,a}^2)d_{e,1} + (1 + 2T_a D_a \omega_{0,a})(c_{e,0} + c_{e,1})}{T_a m_{e,1}}, \\ a_3 &= \frac{(2D_a \omega_{0,a} + T_a \omega_{0,a}^2)m_{e,1} + (1 + 2T_a D_a \omega_{0,a})d_{e,1} + T_a(c_{e,0} + c_{e,1})}{T_a m_{e,1}}, \\ a_4 &= \frac{(1 + 2T_a D_a \omega_{0,a})m_{e,1} + T_a d_{e,1}}{T_a m_{e,1}} \end{aligned}$$

und

$$\begin{aligned} b_0 &= \frac{\omega_{0,a}^2 c_{e,0} c_{e,1}}{T_a m_{e,1}}, \\ b_1 &= \frac{\omega_{0,a}^2 c_{e,0} d_{e,1}}{T_a m_{e,1}} \quad \text{sowie} \\ b_2 &= \frac{\omega_{0,a}^2 c_{e,0}}{T_a} \end{aligned}$$

angegeben werden können. Das entsprechende Zustandsraummodell in Regelungsnormalform lautet

$$\dot{\mathbf{x}}_3^{(k)} = \mathbf{A}_3 \mathbf{x}_3^{(k)} + \mathbf{b}_3 u_3^{(k)} \quad (5.26a)$$

$$f^{(k)} = \mathbf{c}^T \mathbf{x}_3^{(k)} \quad (5.26b)$$

mit der Systemmatrix  $\mathbf{A}_3 \in \mathbb{R}^{(5 \times 5)}$ , dem Eingangsvektor  $\mathbf{b} \in \mathbb{R}^{(5 \times 1)}$  und dem Ausgangsvektor  $\mathbf{c} \in \mathbb{R}^{(5 \times 1)}$ . Das Signal  $u_3^{(k)}$  ist die Stellgröße des Kraft-Momentenreglers,  $\mathbf{x}_3^{(k)}$  der Zustandsvektor und  $f^{(k)}$  die Endeffortkraft bzw. das -moment im entsprechenden Freiheitsgrad  $k$ .

## Reglerauslegung nach dem Betragsoptimum

Der für die Strecke (5.26) gewählte Zustandsregler

$$u_3^{(k)} = \mathbf{r}_3^T \mathbf{x}_3^{(k)} \quad (5.27)$$

wird nach dem Betragsoptimum für Strecken mit Nullstellen nach Papiernik [1986] ausgelegt. Gegenüber den stellgrößenoptimalen Auslegungsvorschriften von Ricatti-Reglern führt dieses Verfahren bei gleicher Stellamplitude auf besser gedämpfte Regelkreise. Wie in dem oben genannten Aufsatz ausgeführt, sind die Nennerkoeffizienten  $k_i$  des geschlossenen Regelkreises durch die Lösung des Gleichungssystems

$$k_1^2 = \frac{b_0^2}{D} \quad (5.28a)$$

$$k_s^2 - 2k_1k_3 = \frac{b_1^2 - 2b_0b_2}{D} \quad (5.28b)$$

$$k_3^2 - 2k_2k_4 + 2k_1k_5 = \frac{b_2^2}{D} \quad (5.28c)$$

$$k_4^2 - 2k_3k_5 + 2k_2 = 0 \quad (5.28d)$$

$$k_5^2 - 2k_4 = 0 \quad (5.28e)$$

bestimmt, wobei der zusätzliche Parameter  $D$  ein Maß für die Dämpfung des Regelkreises ist. Der Rückführungsvektor  $\mathbf{r}_3^{(5 \times 1)}$  lautet

$$\mathbf{r}_3 = [r_i] : r_i = k_i - a_{i-1} \quad \text{für alle } i \in [1; 5]_{\mathbb{N}}. \quad (5.29)$$

Zur Vermeidung einer bleibenden Regelabweichung wird der Filter

$$m_3 = \frac{k_1}{b_0} \quad (5.30)$$

vor das System geschaltet, so dass sich der geschlossene Regelkreis der betragsoptimalen Regelung mit der Sollkraft bzw. dem Sollmoment  $f_{\text{ref}}^{(k)}$  zu

$$\dot{\mathbf{x}}_3^{(k)} = (\mathbf{A}_3 - \mathbf{b}_3 \mathbf{r}_3^T) \mathbf{x}_3^{(k)} + \mathbf{b}_3 m_3 f_{\text{ref}}^{(k)} \quad (5.31a)$$

$$f^{(k)} = \mathbf{c}_3^T \mathbf{x}_3^{(k)} \quad (5.31b)$$

ergibt. Mit dem charakteristischen Polynom  $|s\mathbf{I} - (\mathbf{A}_3 - \mathbf{b}_3 \mathbf{r}_3^T)|$  können die Polstellen des Systems berechnet werden.

Die Auslegung des Reglers wird in dem folgenden Beispiel für die aus den vorherigen Abschnitten bekannte Regelstrecke gezeigt:

**Beispiel 5** Gegeben sei die Regelstrecke (5.25) mit den Parametern  $T_a = 7,48 \text{ ms}$ ,  $\omega_{0,a} = 625,232 \text{ rad/s}$  und  $D_a = 0,81 \text{ /s}$  sowie den Umgebungsparametern  $c_{e,0} = c_{e,1} = 200 \text{ N/mm}$ ,

$m_{e,1} = 50 \text{ kg}$  und  $d_{e,1} = 1000 \text{ kg/s}$ .

Die Lösung des Gleichungssystems (5.28) mit einem vorgegebenen Koeffizienten  $k_4 = a_4 = 1,167 \cdot 10^3$  lautet

$$\begin{aligned} D &= 3,150 \cdot 10^9, & k_1 &= 744,9 \cdot 10^9, \\ k_2 &= 6,338 \cdot 10^9, & k_3 &= 203,9 \cdot 10^6 \quad \text{und} \\ k_4 &= 690,4 \cdot 10^3. \end{aligned}$$

Daraus ergeben sich die Pole des geschlossenen Regelkreises (5.31) zu

$$\begin{aligned} s_{1,2} &= (-10 \pm j62,45) \text{ rad/s}, \\ s_{3,4} &= (-286,66 \pm j492,63) \text{ rad/s} \quad \text{und} \\ s_5 &= -573,25 \text{ rad/s}. \end{aligned}$$

### Dynamischer Zustandsregler

Die Auslegung eines dynamischen Zustandsreglers ist in Föllinger [1990, Kap. 13.8.2] hergeleitet. Durch sie können bleibende Regelabweichungen im stationären Fall, bedingt durch Modellungenauigkeiten, ausgeglichen werden. Dazu werden zu der ursprünglichen Regelstrecke zusätzliche Integratoren und damit zusätzliche Systemzustände hinzugefügt. Deren Anzahl ergibt sich aus der Anzahl der Steuer- und Regelgrößen. In dem hier beschriebenen Fall der Regelstrecke aus Gleichung (5.25) wird ein zusätzlicher Integrator mit der Zustandsvariable  $e^{(k)}$  verwendet. Die Beschreibung der erweiterten Strecke lautet,

$$\begin{bmatrix} \dot{\mathbf{x}}_3^{(k)} \\ \dot{e}^{(k)} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{A}_3 & 0 \\ -\mathbf{c}_3^T & 0 \end{bmatrix}}_{\mathbf{A}_{\text{dyn},3}} \underbrace{\begin{bmatrix} \mathbf{x}_3^{(k)} \\ e^{(k)} \end{bmatrix}}_{\mathbf{b}_{\text{dyn},3}} + \underbrace{\begin{bmatrix} \mathbf{b}_3 \\ 0 \end{bmatrix}}_{\mathbf{b}_{\text{dyn},3}} u_3^{(k)} \quad (5.32a)$$

während der erweiterte Regler die Struktur

$$u_3 = \underbrace{\begin{bmatrix} -\mathbf{r}_{x,3}^T - r_{p,3} \mathbf{c}^T & r_{i,3} \end{bmatrix}}_{\mathbf{r}_{\text{dyn},3}^T} \begin{bmatrix} \mathbf{x}_3^{(k)} \\ e^{(k)} \end{bmatrix} \quad (5.32b)$$

besitzt. Dabei ist  $\mathbf{A}_{\text{dyn},3} \in \mathbb{R}^{(6 \times 6)}$  die Systemmatrix des erweiterten Systems und  $\mathbf{b}_{\text{dyn},3} \in \mathbb{R}^{(1 \times 6)}$  der entsprechende Eingangsvektor. Abbildung 5.7 verdeutlicht die Struktur des dynamischen Zustandsreglers  $\mathbf{r}_{\text{dyn},3} \in \mathbb{R}^{(1 \times 6)}$ : Der Skalar  $r_{i,3}$  bestimmt die Zeitkonstante des zusätzlichen Integrators, während  $r_{p,3}$  die Funktion des Vorfilters übernimmt. Der Vektor  $\mathbf{r}_{x,3}^T$  führt die Systemzustände gewichtet auf den Eingang der Strecke zurück.

Die Pole  $s_1$  bis  $s_5$  der betragsoptimalen Zustandsregelung (5.31) bleiben auch in der dynamischen Zustandsregelung erhalten. Der durch den zusätzlichen Systemzustand hinzugekommene reelle Pol  $s_{\text{dyn}}$  soll ein Ausregeln der bleibenden Regelabweichung in der gleichen zeitlichen Größenordnung bewirken, in der auch der Einregelvorgang der betragsoptimalen Zustandsregelung stattfindet. Die genaue Lage wird unter Berücksichtigung der zu erwartenden Modellfehler in Simulationen ermittelt.

Durch geeignete Methoden der Polvorgabe [Kautsky und Nichols, 1985] wird ein erweiterter Regler  $\mathbf{r}_{\text{dyn},3}$  für die Pole  $s_1$  bis  $s_5$  und  $s_{\text{dyn}}$  sowie die erweiterte Strecke mit dem Matrizen-Vektor-Paar  $(\mathbf{A}_{\text{dyn},3}, \mathbf{b}_{\text{dyn},3})$  berechnet. Die Beschreibung des Systems im Zustandsraum lautet schließlich

$$\begin{bmatrix} \dot{\mathbf{x}}_3^{(k)} \\ \dot{e}^{(k)} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_3 - \mathbf{b}_3(\mathbf{r}_{x,3}^T + r_{p,3}\mathbf{c}_3^T) & \mathbf{b}_3 r_{i,3} \\ -\mathbf{c}_3^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_3^{(k)} \\ e^{(k)} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_3 r_{p,3} \\ 1 \end{bmatrix} f_{\text{ref}}^{(k)} \quad (5.33a)$$

$$f^{(k)} = \begin{bmatrix} \mathbf{c}_3^T & 0 \end{bmatrix} \mathbf{x}_3^{(k)}. \quad (5.33b)$$

Eine stationäre Reglerverstärkung von eins wird erreicht, wenn die Bedingung

$$r_{p,3} = -(\mathbf{c}^T \mathbf{A}^{-1} \mathbf{b})^{-1} \quad (5.34)$$

erfüllt wird.

Das folgende Beispiel erweitert den oben berechneten einfachen Zustandsregler zu einem dynamischen Zustandsregler:

**Beispiel 6** Gegeben sei die in Beispiel 5 beschriebene Regelstrecke und die dort ermittelte betragsoptimale Zustandsregelung. Der zusätzliche, freie Pol der erweiterten Strecke wird auf  $s_{\text{dyn}} := -300 \text{ rad/s}$  gelegt.

Damit ergibt sich eine dynamische Zustandsregelung gemäß der Gleichungen (5.32b) und (5.34) von

$$\mathbf{r}_{x,3} = \begin{bmatrix} -5,1058 & -0,0184 & 509,2115 & 1529 & 0,1463 \end{bmatrix},$$

$$r_{p,3} = 100 \cdot 10^{-3} \quad \text{und}$$

$$r_{i,3} = 0,054.$$

## Beobachter

Da die Zustandsgrößen  $\mathbf{x}_3^{(k)}$  des Parallelroboters und der Umgebung in der Aktionsprimivsteuerung nicht messbar sind, wird ein Luenberger-Beobachter für die Strecke (5.26) entworfen. Das Verfahren der Polvorgabe auf das bekannte Streckenmodell mit den Matrizen bzw. Vektoren  $\mathbf{A}_3$ ,  $\mathbf{b}_3$  und  $\mathbf{c}_3^T$  angewendet, führt auf den Korrekturvektor  $\mathbf{h}_3 \in \mathbb{R}^{(5 \times 1)}$ . Die Zustandsraumdarstellung des dynamischen Zustandsreglers mit Beobachter in dem Frei-

heitsgrad  $k$  lautet dann

$$\begin{bmatrix} \dot{\hat{\mathbf{x}}}_3^{(k)} \\ \dot{e}^{(k)} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_3 - \mathbf{h}_3 \mathbf{c}_3^T - \mathbf{b}_3 \mathbf{r}_{x,3}^T & \mathbf{b}_3 r_{i,3} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_3^{(k)} \\ e^{(k)} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_3 r_{p,3} & \mathbf{h}_3 - \mathbf{b}_3 r_{p,3} \\ 1 & -1 \end{bmatrix} \begin{bmatrix} f_{\text{ref}}^{(k)} \\ f^{(k)} \end{bmatrix} \quad (5.35a)$$

$$u_3^{(k)} = \begin{bmatrix} -\mathbf{r}_{x,3}^T & r_{i,3} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_3^{(k)} \\ e^{(k)} \end{bmatrix} + \begin{bmatrix} r_{p,3} & -r_{p,3} \end{bmatrix} \begin{bmatrix} f_{\text{ref}}^{(k)} \\ f^{(k)} \end{bmatrix}. \quad (5.35b)$$

Wie in Abbildung 5.7 zu sehen, ist der Ausgang des Reglers die Stellgröße  $u_3^{(k)}$ , welche in den Führungsgrößengenerator gespeist wird. Die Eingangsgrößen sind die gemessene Istkraft bzw. das Istmoment  $f^{(k)}$  und die vom aktuellen Aktionsprimitiv vorgegebene Sollgröße  $f_{\text{ref}}^{(k)}$ . Die Implementierung des dynamischen Zustandsreglers wird mit dem folgenden Beispiel vervollständigt:

**Beispiel 7** Gegeben sei der in Beispiel 6 berechnete dynamische Zustandsregler und das Zustandsraummodell der Regelstrecke (5.26) bestehend aus Führungsgrößengenerator und dynamischem Umgebungsmodell. Die vorgegebenen Polstellen des Beobachters richten sich nach den Polen des geschlossenen Regelkreises und sind

$$\begin{aligned} s_1 &:= -140 \text{ rad/s}, & s_2 &:= -160 \text{ rad/s}, \\ s_3 &:= -3000 \text{ rad/s}, & s_4 &:= -3100 \text{ rad/s} \quad \text{sowie} \\ s_5 &:= -800 \text{ rad/s}. \end{aligned}$$

Daraus ergibt sich nach [Kautsky und Nichols, 1985] der Korrekturvektor

$$\mathbf{h}_3 = \begin{bmatrix} -84.485 \cdot 10^{-3} \\ -14.484 \\ -40.797 \cdot 10^{-6} \\ 2.811 \cdot 10^{-6} \\ 9.978 \cdot 10^{-3} \end{bmatrix}.$$

Die Simulationen zur Validierung des dynamischen Zustandsreglers wurden mit dem im Anhang A.1 beschriebenen Parallelroboter HEXA durchgeführt. In Abbildung 5.8(a) ist zunächst die Sprungantwort der dynamischen Umgebung mit den Parametern  $c_{e,0} = c_{e,1} = 200 \text{ N/mm}$ ,  $m_{e,1} = 50 \text{ kg}$  und  $d_{e,1} = 1000 \text{ kg/s}$  dargestellt. Anschließend zeigt Abbildung 5.8(b) die Sprungantwort des dynamischen Zustandsreglers mit Beobachter. Gegenüber dem PI-Regler aus Abschnitt 5.2.2 ist eine deutlich verkürzte Einschwingzeit und geringeres Überschwingen festzustellen.



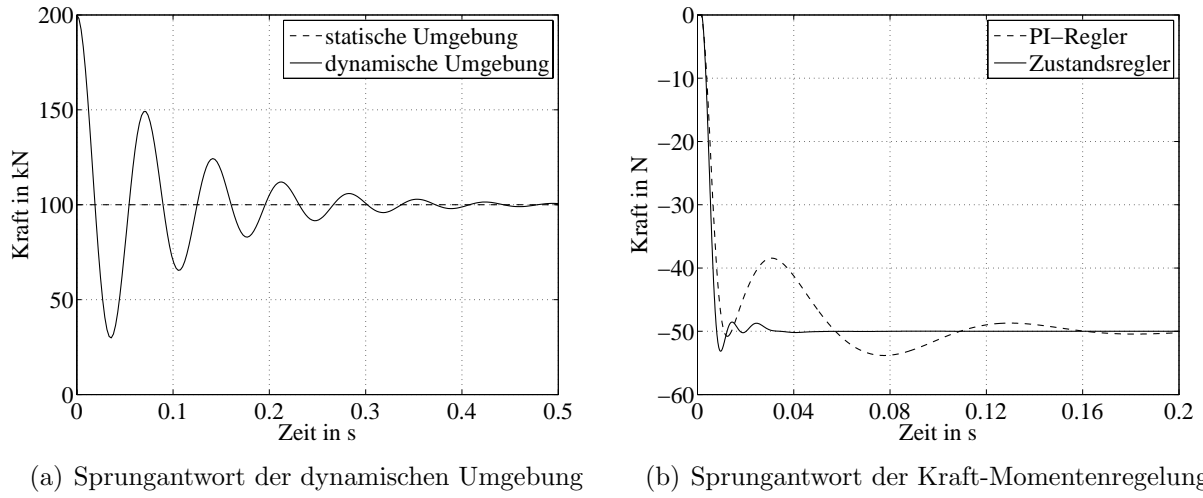


Abb. 5.8: Simulationsergebnisse der Kraft-Momentenregelung mit dynamischen Umgebungsmodell

Der vorgestellte Ansatz des dynamischen Zustandsreglers, ausgelegt nach dem Betragsoptimum, stellt eine Möglichkeit zur Integration des dynamischen Umgebungsmodells dar.

## 5.3 Transitionsplaner

Der Transitionsplaner fasst die Regelungsgesetze zur Kontaktaufnahme bzw. Annäherung des Endeffektors an die Umgebung und zur Dämpfung der Abpralleffekte zusammen.

### 5.3.1 Kontaktaufnahme

Die Kontaktaufnahme dauert vom Beginn des Aktionsprimitivs zur Kontaktregelung zum Zeitpunkt  $t_0$  (siehe Abbildung 1.1) bis zur Detektion des ersten Umgebungskontakts bei  $t_k$ . Während dieser Phase bewegt sich der Endeffektor mit konstanter Geschwindigkeit auf die Umgebung zu. Der Führungsgrößengenerator für einen Freiheitsgrad im Task-Frame ist ein lineares System bestehend aus einem  $PT_2$ -Element mit nachgeschaltetem Integrator. Dessen Zustandsraumdarstellung lautet

$${}^{\text{TF}} \begin{bmatrix} \dot{x}_{\text{ref}} \\ \ddot{x}_{\text{ref}} \\ \dddot{x}_{\text{ref}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -a_1 & -a_2 \end{bmatrix} {}^{\text{TF}} \begin{bmatrix} x_{\text{ref}} \\ \dot{x}_{\text{ref}} \\ \ddot{x}_{\text{ref}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} {}^{\text{TF}} v_{\text{ref}}. \quad (5.36)$$

Dabei ist  ${}^{\text{TF}}v_{\text{ref}}$  die Annäherungsgeschwindigkeit des Endeffektors im Task-Frame. Die Parameter des PT<sub>2</sub>-Elements gehorchen der Bedingung

$$\frac{a_2^2}{4} = a_1, \quad (5.37)$$

so dass sich eine Dämpfung von eins einstellt. Der maximal zulässige Ruck des Parallelroboters sowie die gewünschte Beschleunigung sind weitere Randbedingungen, die berücksichtigt werden müssen.

In dem Zeitintervall  $[t_0; t_k]$  speist der Transitionsplaner in den kontaktgeregelten Freiheitsgraden die Trajektorie

$$T_{\text{ref},a}^{(k)} = \left\{ {}^{\text{TF}}x(t)_{\text{ref}}, {}^{\text{TF}}\dot{x}(t)_{\text{ref}}, {}^{\text{TF}}\ddot{x}(t)_{\text{ref}} \right\} \quad (5.38)$$

in den Task-Frame-Formalismus ein.

### 5.3.2 Dämpfung

Die im transienten Umgebungskontakt auftretenden Reaktionskräfte werden nach Gleichung (4.55) von der Massenträgheitsmatrix beeinflusst. Stelleinwirkungen des Reglers zu den Zeitpunkten  $t_k$  sind wirkungslos. In den Intervallen zwischen den Umgebungskontakten kann die Bewegung des Endeffektors über die Lageschnittstelle nicht direkt beeinflusst werden, da die unterlagerte Antriebsregelung die Stellgröße verzögert. Der Vorwärtspfad vom Eingang bis zur Momentenschnittstelle des Parallelroboters im  $k$ -ten Freiheitsgrad ist

$$F^{(k)}(s) = G_{K_2}^{(k)} G_{K_1}^{(k)} \frac{1}{T_c s + 1} = V_1 V_2 \frac{T_{i,1} T_{r,2} s^2 + (T_{i,1} + T_{r,2}) s + 1}{T_{i,1} T_{l,2} T_c s^3 + T_{i,1} (T_{l,2} + T_c) s^2 + T_{i,1} s} \quad (5.39)$$

und damit ein System dritter Ordnung, das eine entsprechende Verzögerung besitzt. Durch die geeignete Wahl der Solltrajektorie muss gewährleistet werden, dass die unterlagerte Antriebsregelung den Endeffektor möglichst rasch auf der Oberfläche zur Ruhe kommen lässt und ein sicherer Umgebungskontakt gewährleistet ist.

Mit der Detektion des ersten Umgebungskontakts zum Zeitpunkt  $t_k$  wird die bisher gültige Solltrajektorie  $T_{\text{ref},a}^{(k)}$  modifiziert, so dass die Sollposition an der aktuellen Position angehalten und für die gesamte Dauer der Dämpfung konstant gehalten wird. Desweiteren setzt der Algorithmus die Sollgeschwindigkeit bzw. -beschleunigung auf null. Die dann gültige Trajektorie lautet

$$T_{\text{ref},d}^{(k)} = \left\{ {}^{\text{TF}}x \Big|_{t=t_k}, 0, 0 \right\} \quad \forall t \in [t_k; t_d]. \quad (5.40)$$

In diesem Fall stabilisiert die unterlagerte Antriebsregelung den Parallelroboter im Umgebungskontakt.

Die Simulation des transienten Umgebungskontakts mit Moreaus Stoßprozess validiert die obige Aussage. Die Bewegungsgleichung des Parallelroboters im kartesischen Roboterkoordinatensystem der Form (4.51) lauten

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}} - \mathbf{p} = \mathbf{r}. \quad (5.41)$$

Das Vektorfeld  $\mathbf{p}$  fasst neben der Matrix der Coriolis- und Zentrifugalkräfte  $\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}})$  und dem Vektor der Gravitationskräfte  $\mathbf{g}(\mathbf{x})$ , die Elemente der Umrichterverzögerung (3.3) sowie der unterlagerten Antriebsregelung (3.5) bzw. (3.6) mit dem inversen dynamischen Modell (3.14) zusammen. Es kann daher als Funktion der Antriebswinkel, Winkelgeschwindigkeiten und der Solltrajektorie

$$\mathbf{p}(\mathbf{h}(\mathbf{q}), \mathbf{J}_x^{-1}\dot{\mathbf{q}}, \mathbf{x}_{\text{ref}}, \dot{\mathbf{x}}_{\text{ref}}, \ddot{\mathbf{x}}_{\text{ref}})$$

formuliert werden. Die Notation der Istwerte in Gelenkkoordinaten soll verdeutlichen, dass die unterlagerte Antriebsregelung in der Simulation ebenfalls in Gelenkkoordinaten ausgeführt wird. Mit der definierten Solltrajektorie  $T_{\text{ref},d}$  vereinfacht sich die Bewegungsgleichung (5.40) zu

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}} - \mathbf{p}(\mathbf{h}(\mathbf{q}), \mathbf{J}_x^{-1}\dot{\mathbf{q}}, \mathbf{x}_{\text{ref}}, \mathbf{0}, \mathbf{0}) = \mathbf{r}. \quad (5.42)$$

Die anschließende Normierung mit der inversen Massenträgheitsmatrix  $\mathbf{M}^{-1}$  führt schließlich auf die Normalform, die in der Simulation nach Moreaus Stoßprozess verwendet werden kann:

$$\ddot{\mathbf{x}} - \tilde{\mathbf{p}}(\mathbf{h}(\mathbf{q}), \mathbf{J}_x^{-1}\dot{\mathbf{q}}, \mathbf{x}_{\text{ref}}, \mathbf{0}, \mathbf{0}) = \tilde{\mathbf{r}} \quad \text{mit} \quad \tilde{\mathbf{p}} = \mathbf{M}^{-1}\mathbf{p} \quad \text{und} \quad \tilde{\mathbf{r}} = \mathbf{M}^{-1}\mathbf{r}.$$

Der unilaterale Umgebungskontakt ist eine Oberfläche in der  $x$ - $y$ -Ebene und sei durch

$$\Gamma(\mathbf{x}) = -z - z_0 = 0 \quad (5.43)$$

beschrieben, so dass sich der Arbeitsraum zu

$$L = \{\mathbf{x} \in \mathbb{R}^n : \Gamma(\mathbf{x}) \leq 0\} \quad (5.44)$$

ergibt. Die Projektion des *Catching-up-Algorithmus* in einer Dimension nach Gleichung (4.73) gilt analog auch für  $n$  Freiheitsgrade. Für einen Parallelroboter mit sechs

Freiheitsgraden und der oben beschriebenen Umgebungsgrenze lautet sie

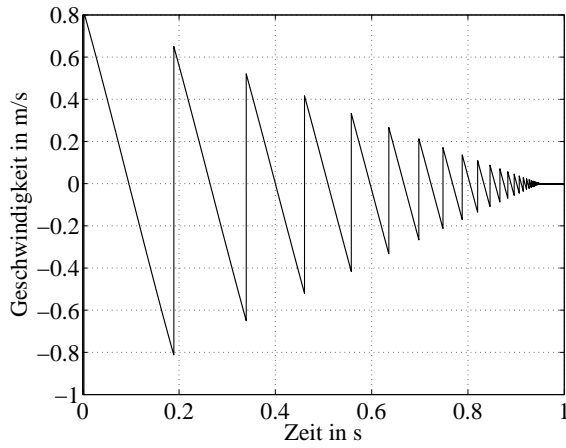
$$\mathbf{w}_{i+1}^v = \begin{cases} \mathbf{u}_i^v + (1-k)h\tilde{\mathbf{p}}_{i+1} : z > z_0 \\ \begin{bmatrix} u_{x,i}^v + (1-k)h\tilde{p}_{x,i+1} \\ u_{y,i}^v + (1-k)h\tilde{p}_{y,i+1} \\ 0 \\ u_{\varphi,i}^v + (1-k)h\tilde{p}_{\varphi,i+1} \\ u_{\theta,i}^v + (1-k)h\tilde{p}_{\theta,i+1} \\ u_{\psi,i}^v + (1-k)h\tilde{p}_{\psi,i+1} \end{bmatrix} : z \leq z_0. \end{cases} \quad (5.45)$$

Die Simulation wurde mit dem dynamischen Modell des Parallelroboters HEXA durchgeführt, das im Anhang A.1 beschrieben wird. Die im Aktionsprimitiv definierte Regelungsaufgabe beschreibt eine Kontaktregelung in der  $z$ -Achse des Roboterkoordinatensystems, während die übrigen Freiheitsgrade lagegeregelt sind. In Abbildung 5.9 sind die Ergebnisse dargestellt. Für alle Simulationen wurde eine Stoßzahl von  $e = 0,8$  angenommen. Der Parallelroboter nähert sich immer mit einer Geschwindigkeit von  $\dot{z} = -1 \text{ m/s}$  der Oberfläche.

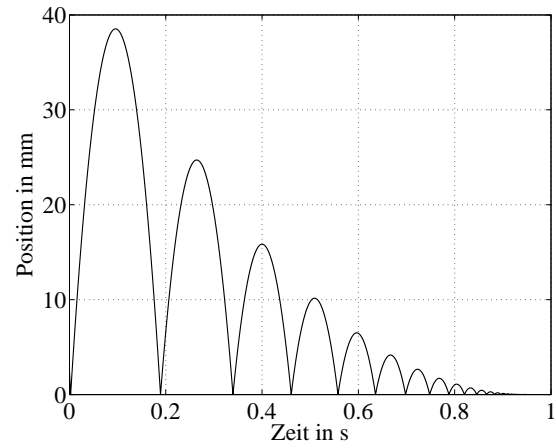
Ein unregelter Parallelroboter, der mit  $\dot{z} = -1 \text{ m/s}$  auf die Oberfläche trifft, verhält sich wie der weiter oben beschriebene Ball. Der Verlauf der Geschwindigkeit in Abbildung 5.9(a) und der Position auf der  $z$ -Achse in Abbildung 5.9(b) stimmt mit den Ergebnissen in Abschnitt 4.3.2 überein.

Wie in den folgenden Kurven zu sehen, beeinflusst die unterlagerte Antriebsregelung die Bewegung zwischen den Umgebungskontakten. Dadurch erreicht der Endeffektor schneller wieder die Oberfläche. Ein wesentlicher Aspekt des Systemverhaltens hängt von der Abtastzeit  $T_s$  ab. Ist diese klein, kann die unterlagerte Antriebsregelung den Endeffektor im Freiraum abbremsen und er trifft mit einer geringeren Geschwindigkeit auf die Oberfläche als diejenige mit der er sie verlassen hat. Dieses Verhalten ist für  $T_s = 10 \mu\text{s}$  in Abbildung 5.9(c) zu beobachten. Nachdem der Endeffektor die Oberfläche das erste Mal mit  $\dot{z} = -1 \text{ m/s}$  berührt, verlässt er sie mit  $\dot{z} = 0,8 \text{ m/s}$ . Die zweite Berührung findet bei  $\dot{z} = -0,66 \text{ m/s}$  statt. Der Endeffektor wurde um  $0,14 \text{ m/s}$  abgebremst. Dadurch ist es möglich auch bei elastischen Stößen mit  $e = 1$  einen sicheren Umgebungskontakt herzustellen. Die in Abbildung 5.9(c) dargestellte Position auf der  $z$ -Achse zeigt, dass der Endeffektor nach ca.  $17 \text{ ms}$  auf der Oberfläche ruht.

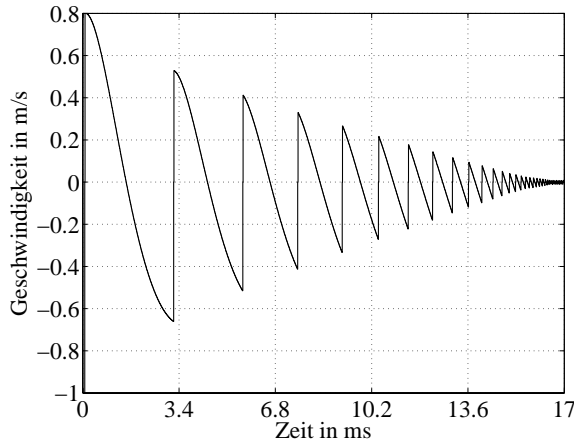
Ist die Abtastzeit zu groß, kann der unterlagerten Antriebsregler den Endeffektor zwischen den Umgebungskontakten nicht mehr abbremsen. Die Energie im System wird nur noch durch die teilelastischen Stöße abgebaut. Abbildung 5.9(e) und (f) zeigen diesen Fall für eine Abtastzeit von  $T_s = 125 \mu\text{s}$ . Zum Zeitpunkt der zweiten Berührung des Endeffektors mit der Umgebung beträgt die Geschwindigkeit  $\dot{z} = -0,83 \text{ m/s}$ . Die Absorption der Bewegungsenergie durch die Umgebungskontakte führt schließlich dazu, dass der Endeffektor nach ca.  $38 \text{ ms}$  auf der Oberfläche ruht.



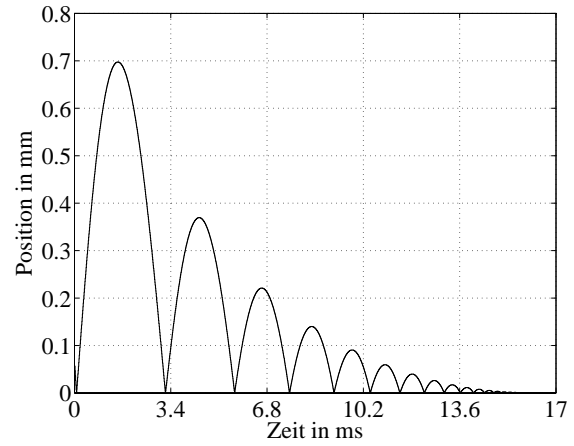
(a) Ungeregelter Parallelroboter HEXA: Geschwindigkeit in der  $z$ -Achse



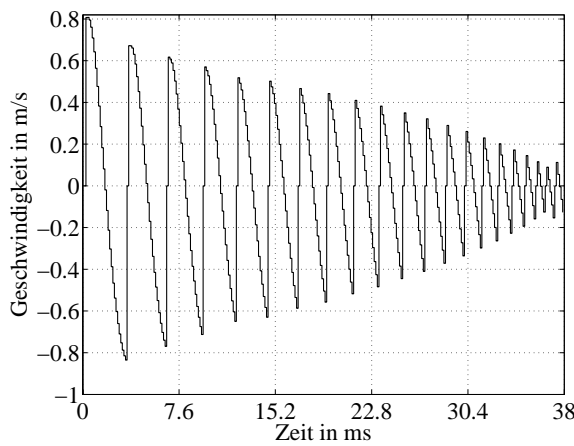
(b) Ungeregelter Parallelroboter HEXA: Position auf der  $z$ -Achse



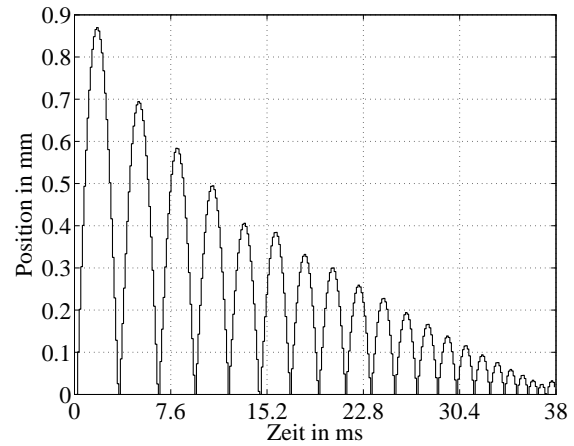
(c) Geregelter Parallelroboter HEXA mit Abtastzeit  $T_s = 10 \mu\text{s}$ : Geschwindigkeit in der  $z$ -Achse



(d) Geregelter Parallelroboter HEXA mit Abtastzeit  $T_s = 10 \mu\text{s}$ : Position auf der  $z$ -Achse



(e) Geregelter Parallelroboter HEXA mit Abtastzeit  $T_s = 125 \mu\text{s}$ : Geschwindigkeit in der  $z$ -Achse



(f) Geregelter Parallelroboter HEXA mit Abtastzeit  $T_s = 125 \mu\text{s}$ : Position auf der  $z$ -Achse

Abb. 5.9: Simulation der Transitionsphase mit Moreaus Stoßprozess

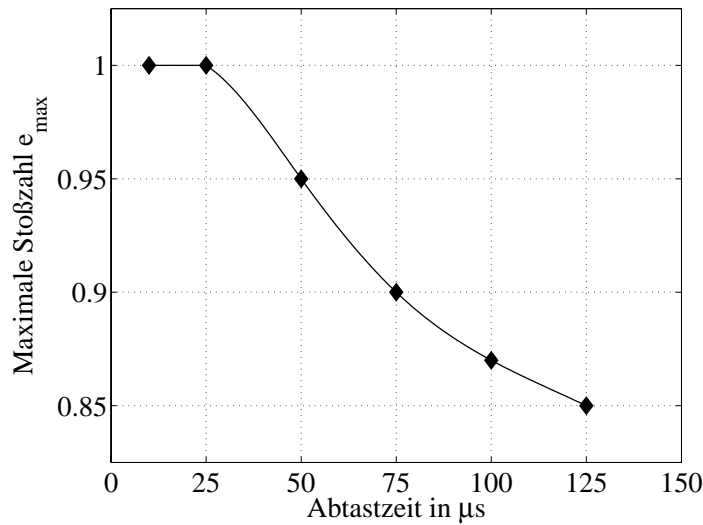


Abb. 5.10: Postulierte maximal beherrschbare Stoßzahl  $e$  in Abhängigkeit von der Abtastzeit

Ausgehend von den oben beschriebenen Simulationsergebnissen wird die in Abbildung 5.10 dargestellte Abhängigkeit der maximalen Stoßzahl  $e_{\max}$  von der Abtastzeit der unterlagerten Antriebsregelung *postuliert*. Ab einer Abtastzeit kleiner  $25 \mu\text{s}$  kann die unterlagerte Antriebsregelung den elastischen Aufprall des Endeffektors auf eine Umgebung dämpfen. Für alle darüber liegenden Abtastzeiten muss die maximale Stoßzahl  $e_{\max}$  berücksichtigt werden.

## 5.4 Stabilität und Sicherheit

Die Stabilität und die Sicherheit der Kontaktregelung ist von mehreren Teilen der Implementierung abhängig. Durch eine geeignete Aufgabenplanung muss der Anwender im Hinblick auf die Montageaufgabe und Umgebungsgeometrie für die konsistente Konfiguration der Steuerung sorgen. Die übergeordnete Verarbeitung der Aktionsprimitive findet in der Robotersteuerung statt, die deren Sicherheit gewährleisten muss. Schließlich müssen die in den Bewegungsmodulen implementierten Regelungen asymptotisch stabil sein, was für die Kontaktregelung in den beschränkten Freiheitsgraden des Systems gezeigt wird.

### 5.4.1 Aufgabenplanung

Die hybride Regelung ermöglicht es für jede Koordinatenachse eines zuvor definierten Task-Frames eine separate Regelungsaufgabe zu definieren. Dies eröffnet dem Anwender eine große Flexibilität, mit der allerdings auch Nachteile einhergehen. In den Publikationen [Vukobratovic und Tuneski, 1994; Yoshikawa, 2000] und nicht zuletzt in diversen Dissertationen [Müller, 1993] wurde darauf hingewiesen, dass eine mit der Umgebungsgeometrie in Widerspruch stehende Task-Frame-Definition zu unbefriedigendem Regelungsverhalten füh-

ren kann. Danach muss die Regelungsaufgabe durch die geeignete Wahl des Task-Frames immer so definiert werden, dass die Regelziele in den Koordinatenachsen voneinander unabhängig oder *orthogonal* sind. *Verkoppelte* Regelziele verursachen lange Einschwingvorgänge oder sogar Instabilitäten.

Im folgenden Abschnitt werden die Auswirkungen der mit Aktionsprimitiven definierten hybriden Kontaktregelungsaufgaben auf die Stabilität untersucht. Die Grundlage der Betrachtungen stellt die in Abschnitt 4.2.2 entwickelte Beschreibung der Kräfte und Momente im Task-Frame dar. Das Ziel der Analyse ist es, eine geeignete Beschreibung der im Task-Frame auftretenden Kräfte und Momente zu finden und daraus Kriterien für eine zuverlässige und schnelle Kontaktregelung mit orthogonalen Regelzielen zu erarbeiten.

### Bahnplanung des Task-Frames

Die zu regelnden Kräfte und Momente sind nach den Bestimmungsgleichungen (4.13) und (4.18) dann vom Bahnverlauf unabhängig, wenn sich die Orientierung des Task-Frames und die des Kontaktkoordinatensystems um die gleiche Achse im Roboterkoordinatensystem und im gleichen Maße ändert. Diese Bedingungen sind im folgenden Korollar zusammengefasst.

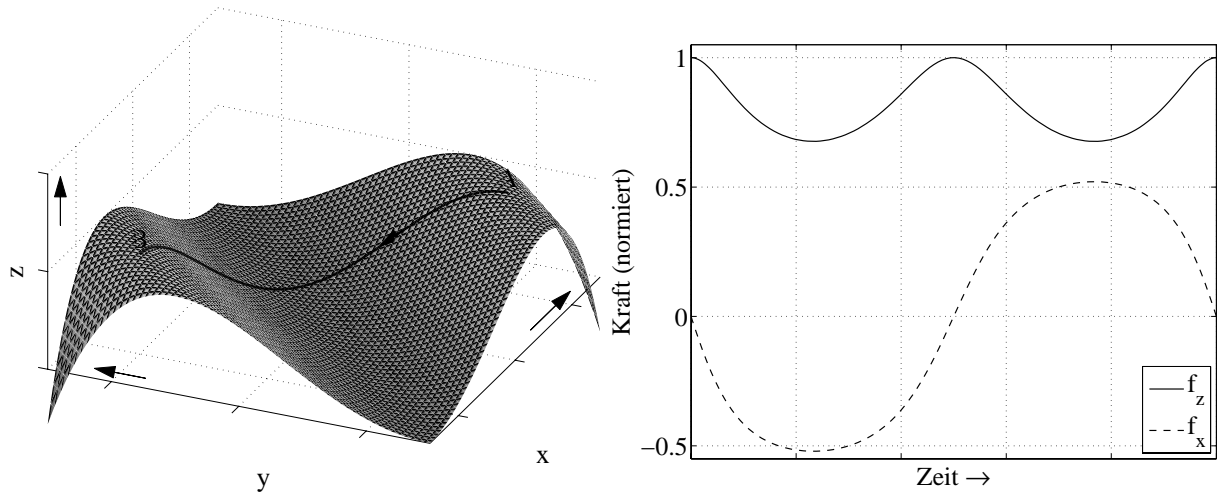
**Korollar 5 (Bahnverlauf des Task-Frames)** *Die Kräfte im Task-Frame eines Aktionsprimitivs  ${}^{\text{TF}}\zeta$  sind dann unabhängig von dem Verlauf der Orientierung des Kontaktpunkts  ${}^{\text{BF}}\mathbf{R}_{\text{IF}}(t)$  und dem Verlauf der Orientierung des Task-Frames  ${}^{\text{BF}}\mathbf{R}_{\text{TF}}(t)$ , wenn*

$${}^{\text{BF}}\mathbf{R}_{\text{TF}}^T(t) {}^{\text{BF}}\mathbf{R}_{\text{IF}}(t) = \text{const} \quad \forall t \geq t_0 \quad (5.46)$$

*gilt.*

Zur Verdeutlichung des Einflusses der Orientierung ist der normierte Kraftverlauf während einer Endeffektorbewegung entlang einer unebenen Oberfläche in den Abbildungen 5.11 dargestellt. Abbildung 5.11(a) zeigt die in den Task-Frame transformierte gewellte Oberfläche auf der sich der Endeffektor im Umgebungskontakt von Punkt A nach B bewegt. Im Laufe der Bewegung ändert sich der Normalenvektor nach Gleichung (4.6a) und mit ihm die Richtung der Endeffektorkraft. Der normierte Kraftverlauf ist in Abbildung 5.11(b) dargestellt. Am Beginn und am Ende der Endeffektorbewegung ist der Normalenvektor parallel zu der  $z$ -Achse des Task-Frames. Die Endeffektorkraft besitzt nur eine  $z$ -Komponente. Ausschließlich durch die Bewegung entlang der geplanten Bahn verändert sich diese Kraftkomponente. Zusätzlich treten Endeffektorkräfte und -momente in den anderen Freiheitsgraden des Koordinatensystems auf, die hier durch den Kraftverlauf in der  $x$ -Achse des Task-Frames veranschaulicht werden. Diese Einflüsse der Umgebungsgeometrie auf die Regelgrößen müssen in der Aufgabenplanung berücksichtigt werden.

In der Praxis bedeutet das, dass ein ortsfest definierter Task-Frame gemäß Fall 1 in Abschnitt 4.2.2 nur bei einer ebenen Umgebungsgeometrie eingesetzt werden kann. Dann ändert



(a) Oberfläche im Task-Frame mit Endeffektorbahn von A nach B

(b) Normierter Kraftverlauf im Task-Frame

Abb. 5.11: Kraftverlauf einer Endeffektorbewegung auf unebener Oberfläche bei konstanter Orientierung des Task-Frames

sich die Orientierung des Kontaktkoordinatensystems im Verlauf des Aktionsprimitivs nicht und die oben genannte Bedingung wird erfüllt.

Der triviale Fall der Bahnplanung tritt ein, wenn  ${}^{\text{BF}}\mathbf{R}_{\text{TF}} := {}^{\text{BF}}\mathbf{R}_{\text{IF}}$ . Dann ist

$${}^{\text{TF}}\boldsymbol{\zeta} = {}^{\text{BF}}\mathbf{R}_{\text{IF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{TF}}\boldsymbol{\zeta}$$

und mit  ${}^{\text{BF}}\mathbf{R}_{\text{IF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} = \mathbf{I}$  gilt  ${}^{\text{TF}}\boldsymbol{\zeta} = {}^{\text{IF}}\boldsymbol{\zeta}$ . Auch diese Konfiguration hat eine Entsprechung in der Praxis. Wird der Task-Frame mit einem roboterinternen Koordinatensystem verankert, so z. B. mit der Roboterhand, dann folgt die Orientierung, eine entsprechende Definition der Regelziele in den übrigen Koordinatenachsen vorausgesetzt, der des Kontaktkoordinatensystems.

### Reglerauswahl in den Koordinatenachsen

Neben der Bahnplanung des Task-Frames verdient die Reglerauswahl in den Koordinatenachsen Beachtung. Die Abhängigkeit der generalisierten Kräfte im Task-Frame ist in dem folgenden Korollar beschrieben.

**Korollar 6 (Abhängigkeit der generalisierten Kräfte im Task-Frame)** *In Übereinstimmung mit den in Gleichung (4.13) und (4.18) hergeleiteten Transformationen der Umgebungskräfte wird die Abhängigkeit der generalisierten Kräfte im Task-Frame durch die*



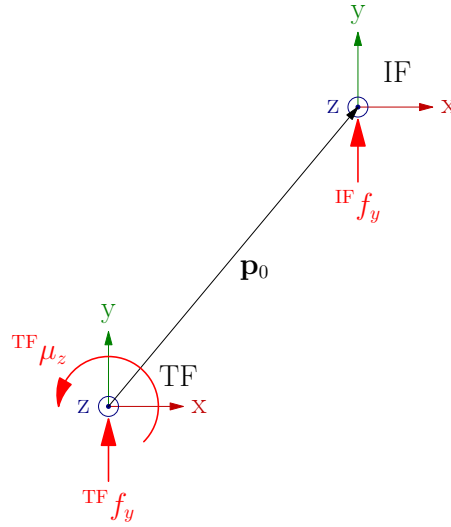


Abb. 5.12: Kräfte und Momente im Task-Frame in Abhängigkeit der Endeffektorkraft im Kontaktkoordinatensystem

*Beziehung*

$${}^{\text{TF}}\mathbf{f} = {}^{\text{TF}}\begin{bmatrix} \zeta \\ \boldsymbol{\mu} \end{bmatrix} = {}^{\text{TF}}\begin{bmatrix} f_x \\ f_y \\ f_z \\ \mu_x \\ \mu_y \\ \mu_z \end{bmatrix} = {}^{\text{TF}}\begin{bmatrix} f_x \\ f_y \\ f_z \\ p_{0,y}f_z - p_{0,z}f_y \\ p_{0,z}f_x - p_{0,x}f_z \\ p_{0,x}f_y - p_{0,y}f_x \end{bmatrix} \quad (5.47)$$

beschrieben. Dabei werden die Kräfte im Task-Frame wie folgt definiert

$${}^{\text{TF}}\boldsymbol{\zeta} = {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{R}_{\text{IF}} {}^{\text{IF}}\boldsymbol{\zeta} = {}^{\text{TF}}\begin{bmatrix} f_x & f_y & f_z \end{bmatrix}^T. \quad (5.48)$$

Die Momente in den rotatorischen Achsen lauten  ${}^{\text{TF}}\mu_x$ ,  ${}^{\text{TF}}\mu_y$  bzw.  ${}^{\text{TF}}\mu_z$  und die Hebelarme sind

$${}^{\text{TF}}\mathbf{p}_0 = {}^{\text{BF}}\mathbf{R}_{\text{TF}}^T {}^{\text{BF}}\mathbf{p}_0 = {}^{\text{TF}}\begin{bmatrix} p_{0,x} & p_{0,y} & p_{0,z} \end{bmatrix}^T. \quad (5.49)$$

Eine Kraft  ${}^{\text{IF}}f_y$  im Kontaktkoordinatensystem verursacht, neben der Kraft  ${}^{\text{TF}}f_y$ , in maximal zwei rotatorischen Achsen des Parallelroboters die Drehmomente  ${}^{\text{TF}}\mu_y$  und  ${}^{\text{TF}}\mu_z$ . Die Situation ist vereinfacht in der zweidimensionalen Abbildung 5.12 dargestellt. Durch das Fehlen einer Ausdehnung des Hebelarms  $\mathbf{p}_0$  in  $z$ -Richtung des Task-Frames tritt ausschließlich das Drehmoment  ${}^{\text{TF}}\mu_z$  auf. Umgekehrt setzt das Vorhandensein eines Drehmoments  ${}^{\text{TF}}\mu_x$  im Task-Frame die zwei Kräfte  ${}^{\text{IF}}f_y$  und  ${}^{\text{IF}}f_z$  im Kontaktkoordinatensystem voraus.

Ein unilateraler und reibungsfreier Umgebungskontakt erlaubt die Definition mehrerer Kontaktregler in einem Aktionsprimitiv nur dann, wenn die Bedingungen der Gleichung (5.47) eingehalten werden und der Bahnverlauf des Task-Frames nach der in Korollar 5 formulierten Bedingung (5.46) geplant ist. Kann die Bedingung (5.46) nicht eingehalten werden, so sind die durch den Umgebungskontakt eingekoppelten Kräfte bzw. Momente Störgrößen, die die Regler in den jeweiligen Koordinatenachsen ausregeln müssen.

### 5.4.2 Robotersteuerung auf der Basis von Aktionsprimitiven

Durch die in Kapitel 2 beschriebenen Funktionsprinzipien der Robotersteuerung wird sichergestellt, dass in den beiden Phasen der Abarbeitung eines Aktionsprimitivs eine vollständige Solltrajektorie in den Task-Frame-Formalismus eingespeist wird. Die Steuerung sorgt weiterhin für die Initialisierung der benötigten Bewegungsmodule mit dem aktuellen Systemzustand, so dass eine sichere Umschaltung zwischen verschiedenen Konfigurationen möglich ist.

### 5.4.3 Stabilität der Kontaktregelung

Bei der in dieser Arbeit vorgestellten Kontaktregelung von Parallelrobotern auf der Basis von Aktionsprimitiven handelt es sich um eine Kaskadenregelung deren äußerster Regelkreis je nach Systemzustand in der Struktur und in seinen Parametern verändert wird. In Abhängigkeit davon, ob sich der Endeffektor im Freiraum, in der Transitionsphase oder im Umgebungskontakt befindet, wird das Regelungsgesetz von einem endlichen Automaten umgeschaltet. Die Stabilität der Kontaktregelung hängt von allen in den Kapiteln 3 bis 5 beschriebenen Komponenten ab.

**Theorem 2 (Stabilität der Kontaktregelung)** *Der endliche Automat (5.13) der struktumschaltenden Kontaktregelung mit dem PI-Kontaktregler*

$$G_{K_3}^{(k)}(s) := V_3^{(k)} \frac{T_{i,3}^{(k)} s + 1}{T_{i,3}^{(k)}}$$

bzw. dem dynamischen Zustandsregler mit Beobachter

$$\begin{aligned} \begin{bmatrix} \dot{\hat{\mathbf{x}}}_3^{(k)} \\ \dot{e}^{(k)} \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_3 - \mathbf{h}_3 \mathbf{c}_3^T - \mathbf{b}_3 \mathbf{r}_{x,3}^T & \mathbf{b}_3 r_{i,3} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_3^{(k)} \\ e^{(k)} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_3 r_{p,3} & \mathbf{h}_3 - \mathbf{b}_3 r_{p,3} \\ 1 & -1 \end{bmatrix} \begin{bmatrix} f_{\text{ref}}^{(k)} \\ f^{(k)} \end{bmatrix} \\ u_3^{(k)} &= \begin{bmatrix} -\mathbf{r}_{x,3}^T & r_{i,3} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_3^{(k)} \\ e^{(k)} \end{bmatrix} + \begin{bmatrix} r_{p,3} & -r_{p,3} \end{bmatrix} \begin{bmatrix} f_{\text{ref}}^{(k)} \\ f^{(k)} \end{bmatrix} \end{aligned}$$

und dem Transitionsplaner mit den Trajektorien  $T_{\text{ref},a}^{(k)}$  und  $T_{\text{ref},d}^{(k)}$  sowie der Führungsgrößen-

generator (5.14) stabilisieren den Parallelroboter

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{J}_x^{-T} \mathbf{B}^F \mathbf{f} = \boldsymbol{\tau}$$

mit der unterlagerten Antriebsregelung nach Gleichung (3.5) und (3.6) während der Freiraumbewegung, der Transitionsphase und im unilateralen Umgebungskontakt in einem Freiheitsgrad  $k \in [1; n]_{\mathbb{N}}$  des Task-Frames, so dass

$$f^{(k)} = f_{\text{ref}}^{(k)} \quad \text{wenn} \quad t \rightarrow \infty.$$

Die übrigen  $n - 1$  Freiheitsgrade des Parallelroboters sind im Verlauf der Kontaktregelung lagegeregelt.

In dem folgenden Beweis wird zunächst die Stabilität der Kontaktregelung in den einzelnen Phasen bewiesen. Anschließend folgt die Stabilität der gesamten Kontaktregelung aus dem Stabilitätsbeweis der Umschaltung zwischen den Reglern.

**BEWEIS (STABILITÄT DER KONTAKTREGELUNG)** Für den Stabilitätsbeweis der Kontaktregelung wird angenommen, dass das dynamische Modell des Parallelroboters im Umgebungskontakt (4.44) sowie das inverse dynamische Modell (3.14) bekannt sind. Die Freiheitsgrade des Parallelroboters sind daher als entkoppelt anzunehmen. Die Ausführungen beziehen sich, mit Ausnahme der Transitionsphase, ohne Einschränkung der Allgemeinheit auf den beschränkten Freiheitsgrad  $k$ . Weiterhin wird angenommen, dass sich der Parallelroboter ausschließlich in einem unilateralen Umgebungskontakt befindet. Dann gelten die folgenden Überlegungen:

- (1) Während der *Kontaktaufnahme* folgt der Endeffektor im Freiheitsgrad  $k$  der durch den Führungsgrößengenerator in Gleichung (5.36) vorgegebenen Trajektorie  $T_{\text{ref},a}^{(k)}$ . Diese berücksichtigt die Randbedingungen des Systems wie Arbeitsraumgrenzen und maximale Belastbarkeit der Parallelroboterstruktur. Die Stabilität dieser Phase ist damit eine direkte Folge der Stabilität der unterlagerten Antriebsregelung, die in Korollar 2 bewiesen ist.
- (2) Während der *Dämpfung* speist die Kontaktregelung die in Gleichung (5.40) beschriebene Trajektorie  $T_{\text{ref},d}^{(k)}$  in den Task-Frame-Formalismus und letztendlich in die unterlagerte Antriebsregelung ein. Die Stabilität der Kontaktaufnahme ist durch die in Abschnitt 5.3.2 beschriebene Simulation mit Moreaus Stoßprozess validiert.
- (3) Der Stabilitätsbeweis der Kraft- und Momentenregelung im *Umgebungskontakt* basiert auf der Stabilität der unterlagerten Antriebsregelung (vgl. Korollar 2), die die Dynamik der nichtlinearen Strecke kapselt und nach außen einheitliche Grenzen der Dynamik für alle Lagen des Endeffektors im Arbeitsraum garantiert. Nach dem Korollar 3 zur Modellfolgeregelung kann die Regelstrecke der Kontaktregelung dann als Reihenschaltung aus dem Führungsgrößengenerator (5.14) und dem Umgebungsmodell angesehen werden.

Der PI-Kraft-Momentenregler für ein statisches Umgebungsmodell nach Gleichung (4.44) und (4.34) wird mit Gleichung (5.21) nach dem Kriterium des Phasenrands ausgelegt. Das Verfahren berücksichtigt implizit das Nyquistkriterium und garantiert so die Stabilität der Regelung.

Der dynamische Zustandsregler mit Beobachter gemäß den Gleichungen (5.35) für ein dynamisches Umgebungsmodell (4.44) bzw. (4.42) berücksichtigt in der Auslegung nach dem Betragsoptimum [Papiernik, 1986] ebenfalls implizit die Stabilität des geschlossenen Regelkreises. Gleiches gilt für die Erweiterung zum dynamischen Zustandsregler. Die Pole des Beobachters können unabhängig von denen des Reglers vorgegeben werden und beeinflussen dessen Stabilität nach dem *Separationstheorem* nicht.

Die Stabilität der Kraft-Momentenvorsteuerung ist in Korollar 4 bewiesen.

- (4) Die *Umschaltung* zwischen den Zuständen  $\mathcal{Z}$  des endlichen Automaten  $EA^{(k)}$  in Definition 2 ist stabil, da es sich bei allen Zustandsübergängen  $\mathcal{U}$  in Tabelle 5.2 um Änderungen der Solltrajektorie der unterlagerten Antriebsregelung handelt. Es wird kein zusätzlicher Regelkreis um die unterlagerte Antriebsregelung geschlossen. Die Stabilität folgt daher direkt aus Korollar 2.

Eine Ausnahme bildet die Zustandsübergangssequenz von der *Anfahrt* in die *Dämpfung* und schließlich zum *Kontakt*. Deren Stabilität ist durch die in Abschnitt 5.3.2 beschriebene Simulation validiert. Der Kraft-Momentenregler befindet sich zu Beginn der Regelung daher sicher im Umgebungskontakt.

Die in dieser Arbeit vorgestellte Kontaktregelung ermöglicht eine sichere und stabile Ausführung von Montageaufgaben mit unilateralem Umgebungskontakt, die mit Robotersteuerungen zur Verarbeitung von Aktionsprimitiven ausgeführt werden.

# Kapitel 6

## Experimentelle Untersuchungen

Die im Folgenden durchgeführten experimentellen Untersuchungen der Kontaktregelung zeigen die Leistungsfähigkeit der vorgestellten Regelungskonzepte in der praktischen Anwendung. Grundlage der Kontaktregelung ist die unterlagerte Antriebsregelung, deren Verhalten während einer Freiraumbewegung analysiert wird. Darauf aufbauend stellen die Experimente das Funktionsprinzip der Kontaktregelung mit der strukturumschaltenden Regelung dar. Die besondere Bedeutung der Aufgabenplanung im Kontext der Aktionsprimitive wird ebenso aufgezeigt wie die Durchführung einer hybriden Regelungsaufgabe.

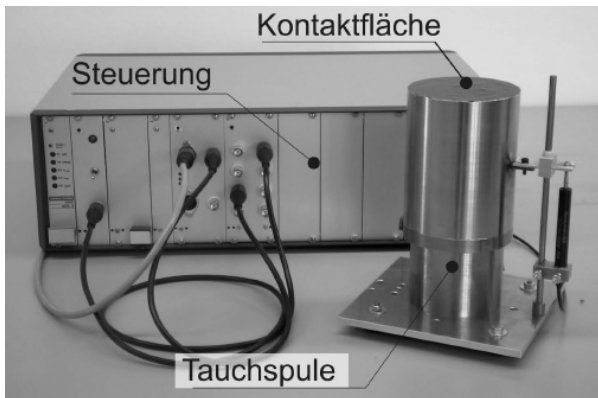
### 6.1 Versuchsaufbau

Der im Anhang A.1 beschriebene Parallelroboter HEXA mit der Aktionsprimitivsteuerung aus Kapitel 2 ist der Versuchsträger der folgenden Experimente. Die Steuerung wurde auf einem PC mit dem Echtzeitbetriebssystem QNX<sup>1</sup> implementiert. In den Umrichtern zur Ansteuerung der Antriebe ist eine Strom- und Geschwindigkeitsregelung integriert. Sie stellen nach außen eine Schnittstelle zur Verfügung, um die vom unterlagerten Antriebsregler berechneten Soll- bzw. Vorsteuergrößen einspeisen zu können. Ein Feldbussystem mit dem *Industrial Automation Protocol* verbindet die Motorumrichter mit dem Steuerungs-PC und überträgt die Soll- und Istwerte der Regelung. Für die Kontaktregelung ist am Endeffektor des Parallelroboters ein sechssachsiger Kraftsensor der Firma JR3 montiert, der ebenfalls über den Feldbus mit der Steuerung verbunden ist.

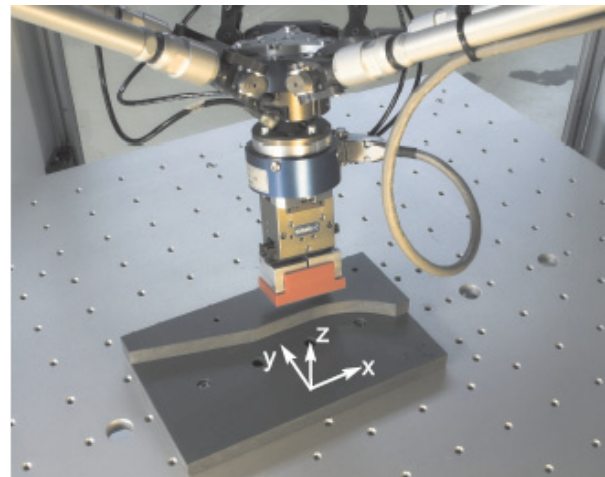
In Abhängigkeit der Aufgabenstellung sind unterschiedliche Konfigurationen des Arbeitsraums möglich: Eine *künstliche Umgebung* erlaubt die Simulation der Kontaktsituation. Das Gerät bildet mit Hilfe eines zustandsgeregelten Tauchspulenaktuators aktiv einen Umgebungskontakt nach. Der Aufbau ist in Abbildung 6.1(a) dargestellt. Die Steuerung enthält einen digitalen Signalprozessor, auf dem der Zustandsregler implementiert ist. Die Lage der Tauchspule wird von einem Potentiometer gemessen und über einen AD-Umsetzer eingelesen. Ein interner Beobachter schätzt die Geschwindigkeit. Die gewünschte Umgebungsteifigkeit des Systems kann an einem Steuer-PC oder an Potentiometern im Bereich von  $c_{e,0\min} = 10 \text{ N/mm}$  bis  $c_{e,0\max} = 25 \text{ N/mm}$  vorgegeben werden. Ebenfalls in der Steuerung

---

<sup>1</sup>QNX ist ein eingetragenes Warenzeichen der Firma QNX Software Systems Ltd.



(a) Künstliche Umgebung



(b) Umgebung mit Kontur zur Simulation der hybriden Regelungsaufgabe

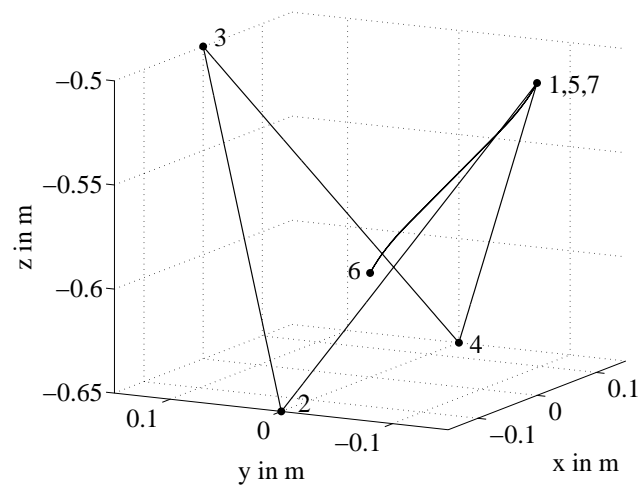
Abb. 6.1: Versuchsaufbau

befindet sich ein Motorverstärker, der das Stellsignal des Reglers verstärkt und den Tauchspulenaktuator mit Energie versorgt.

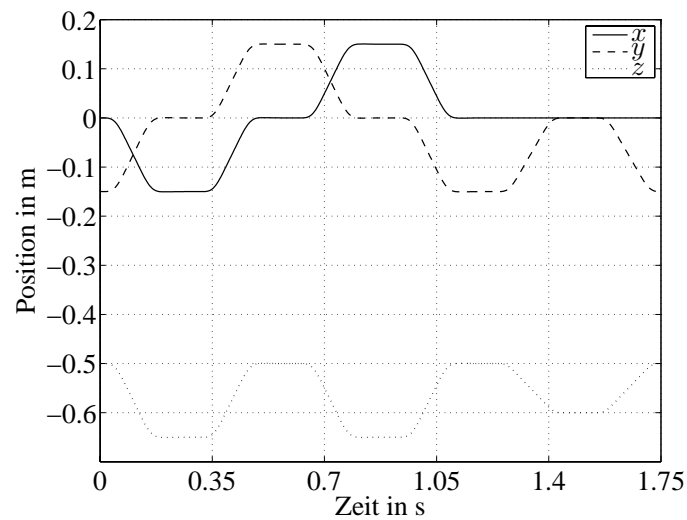
Die Abbildung 6.1(b) zeigt die *Umgebung mit geformter Kontur* zur Simulation hybrider Regelungsaufgaben. Die Umgebung besteht aus Kunststoff und ist fest im Arbeitsraum montiert. An der Arbeitsplattform des Parallelroboters ist ein Kollisionsschutz und der Kraft-Momentensensor montiert. Das Werkstück ist in einem Greifer eingespannt. Eingezeichnet ist außerdem der Task-Frame zu Beginn der hybriden Regelung.

## 6.2 Unterlagerte Antriebsregelung

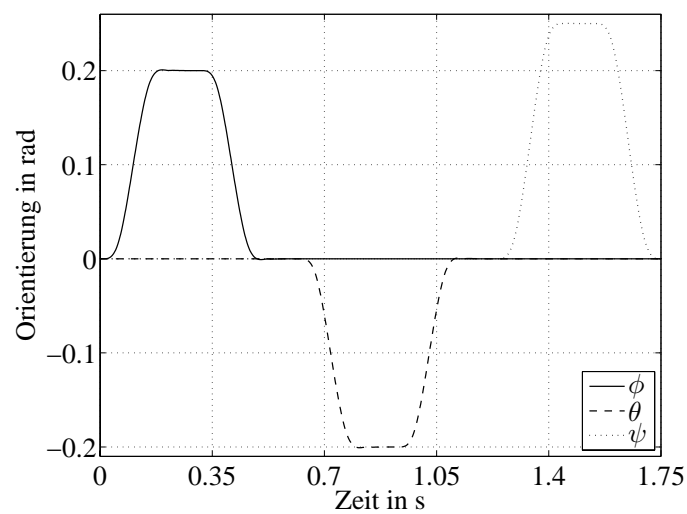
Die geplante Trajektorie in kartesischen Koordinaten ist in den Abbildungen 6.2 zu sehen. Der Endeffektor des Parallelroboters folgt dabei der in Abbildung 6.2(a) dargestellten Bahn um das Zentrum des Arbeitsraums, die so gewählt wurde, dass alle Freiheitsgrade des Parallelroboters ausgenutzt werden. Die kartesische Position als Funktion der Zeit ist in Abbildung 6.2(b) dargestellt, während Abbildung 6.2(c) die Orientierungen zeigt. Die Trajektorie wurde mit einem Planer für Robotersteuerungen auf der Basis von Aktionsprimitiven nach Kolbus u. a. [2005] berechnet, so dass die heuristischen Algorithmen zur Synchronisation der Freiheitsgrade auf eine leicht gekrümmte Bahn zwischen den Wegpunkten (5) und (6) bzw. (6) und (7) führen. Tabelle 6.1 listet die tatsächlich erreichbaren Beschleunigungen und Geschwindigkeiten für einen vorgegebenen Ruck von  $1000 \text{ m/s}^3$  auf.



(a) Endeffektorbahn im Arbeitsraum. Die Bahn wird von den Wegpunkten (1) bis (7) durchfahren.



(b) Sollpositionen des Endeffektors



(c) Sollorientierungen des Endeffektors

Abb. 6.2: Solltrajektorie des Parallelroboters in kartesischen Koordinaten

Parameter	Translation	Rotation
$\dot{\mathbf{x}}_{\max}$	1,4 m/s	2,5 rad/s
$\ddot{\mathbf{x}}_{\max}$	36 m/s <sup>2</sup>	48 rad/s <sup>2</sup>

Tab. 6.1: Parameter der Solltrajektorien

Messungen in Abbildung 6.3 zeigen die Funktionsweise der unterlagerten Antriebsregelung, deren Parameter nach Gleichung (3.11) für eine Dämpfung von  $D = 1$  bestimmt werden. Für den Versuchsträger wurde die Verzögerung der Umrichter von  $T_c = 600 \mu\text{s}$  experimentell ermittelt. In einem Punkt weicht die Implementierung von dem in Kapitel 3 beschriebenen Regelungskonzept ab. Im dem zur Verfügung stehenden Versuchsträger HEXA ist eine Anpassung der Parameter des Geschwindigkeitsreglers zur Laufzeit nicht möglich, da diese im Umrichter berechnet wird. Für den gesamten Arbeitsraum wird die Verstärkung des Geschwindigkeitsreglers daher mit dem durchschnittlichen effektiven Massenträgheitsmoment  $\bar{j}$  ausgelegt. Dieses wird im Zentrum des Arbeitsraums für eine Endeffektormasse von ca.  $m_l = 1 \text{ kg}$  angenähert. Dazu wird die Massenträgheitsmatrix  $\mathbf{M}(\mathbf{x})$  aus dem dynamischen Modell (3.14) durch

$$\mathbf{M}(\mathbf{q}) = \mathbf{J}_x^{-T} \mathbf{M}(\mathbf{x}) \mathbf{J}_x^{-1} \quad (6.1)$$

in Gelenkkoordinaten transformiert und dort der Mittelwert der Hauptdiagonalelemente bestimmt. Das durchschnittliche effektive Massenträgheitsmoment beträgt nach dieser Rechnung  $\bar{j} = 0,21 \text{ kgm}^2$ . Die sich daraus ergebenden Reglerparameter sind in der Tabelle 6.2 aufgeführt.

Parameter	Rechnung	Experiment
$V_1$	131,25	50
$T_{i,1}$	4,8 ms	5 ms
$V_2$	139	139
$T_{r,2}$	2,4 ms	2,4 ms
$T_{l,2}$	4,8 ms	5 ms

Tab. 6.2: Reglerparameter der unterlagerten Antriebsregelung

Zum Vergleich sind die im Experiment verwendeten Einstellungen den Berechnungen in den Beispielen der vorherigen Kapitel gegenübergestellt. Das durch die Umrichter zur Verfügung gestellte Geschwindigkeitssignal hat eine Messverzögerung von  $T_m = 100 \mu\text{s}$ . Die Verzögerung der Datenübertragung durch das verwendete Feldbussystem konnte für die Reglerauslegung nicht vernachlässigt werden. Diese Totzeit wurde zusammen mit der Umrichterverzögerung in der Zeitkonstante  $T_c$  berücksichtigt, die deshalb von den angenommen  $600 \mu\text{s}$  auf  $1,5 \text{ ms}$  ansteigt. Aus den Berechnungen nach den Gleichungen (3.19) und (3.20)



ergeben sich die eingestellten Reglerparameter. Durch den symmetrischen Aufbau des Parallelroboters HEXA bedingt, sind diese für alle sechs Antriebe gleich.

Die Messungen in den untereinander angeordneten Abbildungen 6.3(a), (c) und (e) zeigen das Verhalten der Geschwindigkeitsregelung im unterlagerten Antriebsregler in Gelenkkordinaten, während der Endeffektor in Abbildung 6.2 der Solltrajektorie folgt. Zur besseren Übersicht sind nur die Zustandsgrößen eines Antriebs dargestellt. Leichtes Überspringen in der Sollgeschwindigkeit wird, ebenso wie die oben beschriebene Krümmung der Endeffektorbahn, durch den Trajektorienplaner verursacht. Die maximale Abweichung am Beginn der Trajektorie beträgt ca.  $|\Delta\dot{q}_1| = 0,25 \text{ rad/s}$ , was zu diesem Zeitpunkt 4% Regelabweichung bedeutet.

Die Messungen in den ebenfalls untereinander angeordneten Abbildungen 6.3(b), (d) und (f) zeigen die Soll- und Istwerte sowie die Regelabweichung eines Antriebswinkels. Der Winkelfehler erreicht maximal einen Wert von  $|\Delta q_1| = 0,006 \text{ rad/s}$  oder  $0,34^\circ$ .

Die Vergrößerungen in den Abbildungen 6.3(e) bzw. 6.3(f) zeigen den gleichen Ausschnitt im Intervall von  $t = [1,06 \text{ s}; 1,134 \text{ s}]$ . In Abbildung 6.3(f) ist besonders deutlich die langsamere Taktung der Solltrajektorie gegenüber der Abtastzeit der Lageregelung zu sehen, die durch die unterschiedliche Zykluszeit der Bewegungsmodule und der unterlagerten Antriebsregelung begründet ist.

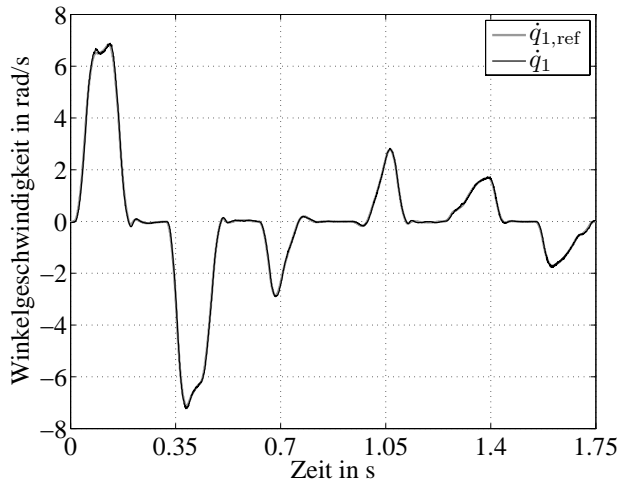
## 6.3 Kontaktregelung

Die Untersuchungen zur Kontaktregelung zeigen zu Beginn die Realisierbarkeit des in Abschnitt 5.1 beschriebenen Konzepts. Anschließend folgt die Diskussion der praktischen Anwendung anhand einer hybriden Regelungsaufgabe. Die oben erwähnten Aspekte der Aufgabenplanung werden abschließend untersucht.

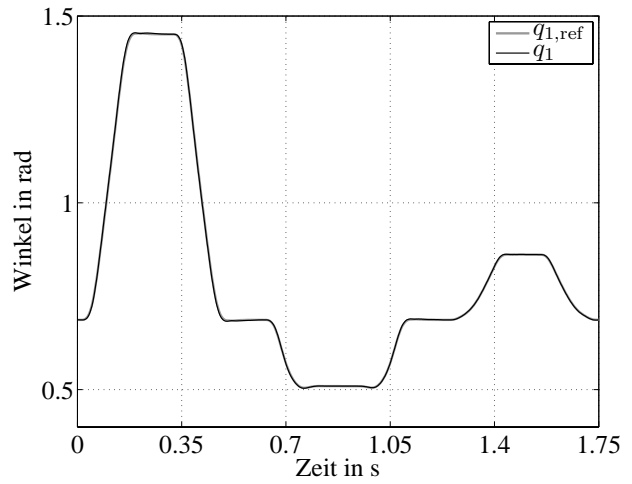
### 6.3.1 Funktionsprinzip

Die Messungen der Zustandsgrößen während der Kontaktaufnahme, der Transitionsphase und der Kraft- bzw. Momentenregelung in Abbildung 6.4 verdeutlichen das Funktionsprinzip der Kontaktregelung. Der Versuchsaufbau mit der *künstlichen Umgebung* gewährleistet einen Umgebungskontakt mit definierten Parametern.

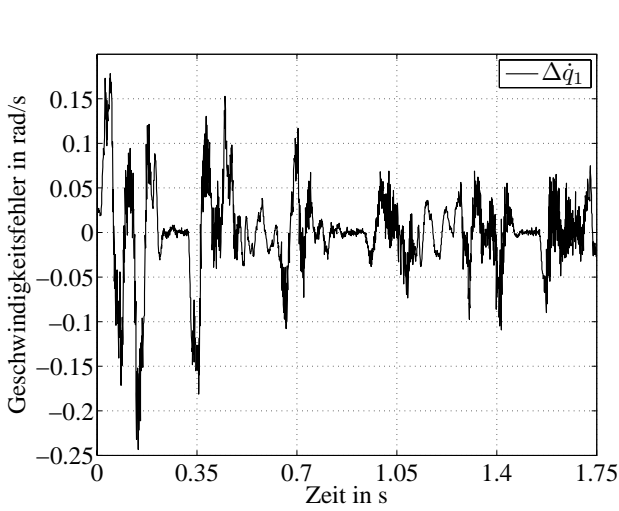
In Abbildung 6.4(a) ist der Verlauf der Endeffektorkraft bei Vorgabe einer Sollkraft von  $f_{z,\text{ref}} = -30 \text{ N}$  mit Vorsteuerung in der  $z$ -Achse dargestellt. Das Regelungskonzept wird anhand der in Abbildung 6.4(b) dargestellten geplanten Trajektorie im Handkoordinatensystem HF, in der die einzelnen Phasen der Kontaktregelung deutlich zu erkennen sind, veranschaulicht: Im Intervall von  $t = [0 \text{ s}; 0,6 \text{ s}]$  befindet sich der Endeffektor noch im Freiraum und ein Geschwindigkeitsplaner ist im Eingriff. Der Endeffektor bewegt sich mit  $\dot{z} = 10 \text{ mm/s}$  auf die Umgebung zu. Mit der Detektion des Umgebungskontakts, zu erkennen an einer kleinen Kraftspitze bei  $t = 0,6 \text{ s}$  in Abbildung 6.4(a), beginnt die Dämpfungsphase. Um



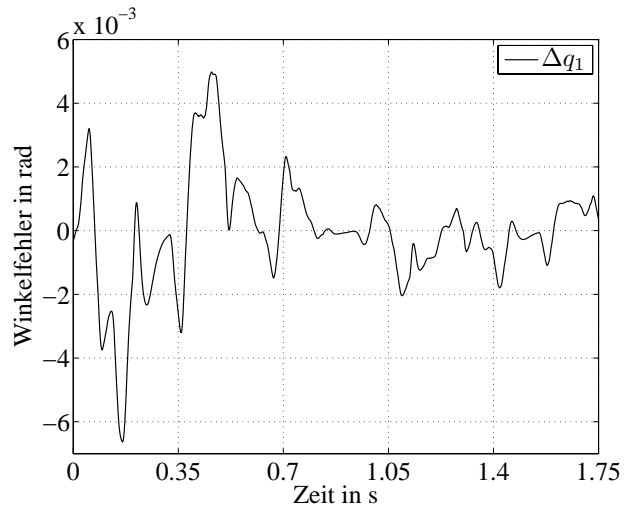
(a) Vergleich der Soll- und Istgeschwindigkeit des ersten Antriebs



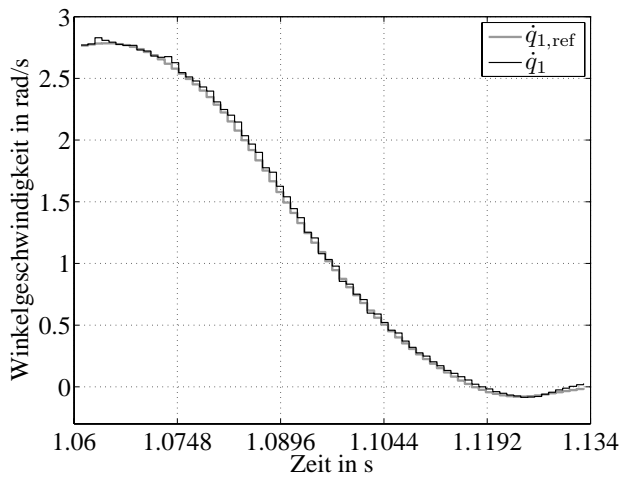
(b) Vergleich der Soll- und Isttrajektorie des ersten Antriebs



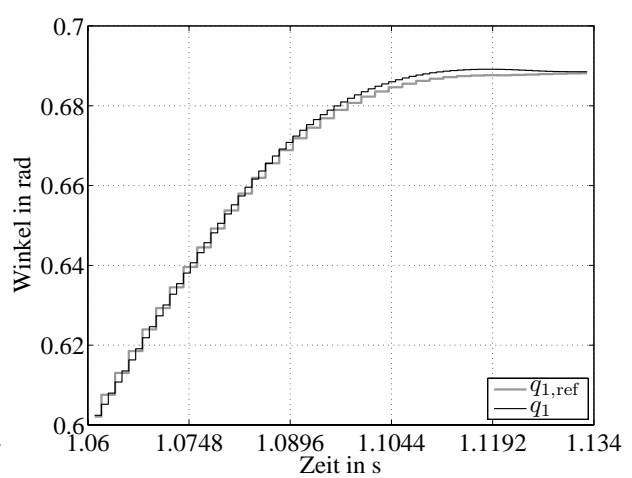
(c) Geschwindigkeitsfehler des ersten Antriebs



(d) Winkelfehler des ersten Antriebs



(e) Vergrößerter Ausschnitt der Soll- und Istgeschwindigkeit des ersten Antriebs



(f) Vergrößerter Ausschnitt der Soll- und Isttrajektorie des ersten Antriebs

Abb. 6.3: Messungen der unterlagerten Antriebsregelung des Parallelroboters HEXA

alle Abprallvorgänge zu unterdrücken und einen sicheren Umgebungskontakt zu gewährleisten, wird der Endeffektor für  $\Delta t = 0,25\text{ s}$  an dem Punkt der *Solltrajektorie* festgehalten, an dem der Umgebungskontakt detektiert wurde. Da die tatsächliche Endeffektorlage der Solltrajektorie immer nacheilt, befindet sich diese Ruhelage innerhalb der Umgebung. Die Endeffektorkraft geht daher nicht vollständig auf null zurück. Dies garantiert einen sicheren Umgebungskontakt. Die Dauer der Dämpfungsphase wird durch eine vorzugebende Zeitkonstante bestimmt, die in diesem Fall mit  $t_{\text{damp}} = 0,25\text{ s}$  zur deutlichen Veranschaulichung der Funktionsprinzipien relativ groß gewählt wurde. Zum Zeitpunkt  $t = 0,85\text{ s}$  beginnt der Kraftregler die gewünschte Endeffektorkraft einzustellen. Die Dynamik der Solltrajektorie ist deutlich an dem steilen Anstieg der Geschwindigkeit in Abbildung 6.4(b) zu erkennen. Dieser wird durch den maximal zulässigen Ruck des Parallelroboters begrenzt. Da in dem Versuchsaufbau die Umgebungssteifigkeit bekannt ist, verhält sich die Kraftregelung wie ein  $\text{PT}_1$ -Element mit einer Zeitkonstante nach Gleichung (5.23). Mit Beginn der Kontaktphase wird auch die Kraftvorsteuerung aufgeschaltet. Hier lässt sich deutlich die Übereinstimmung der in Abbildung 6.4(a) als graue Linie dargestellten Vorsteuerung mit der gemessenen Endeffektorkraft erkennen.

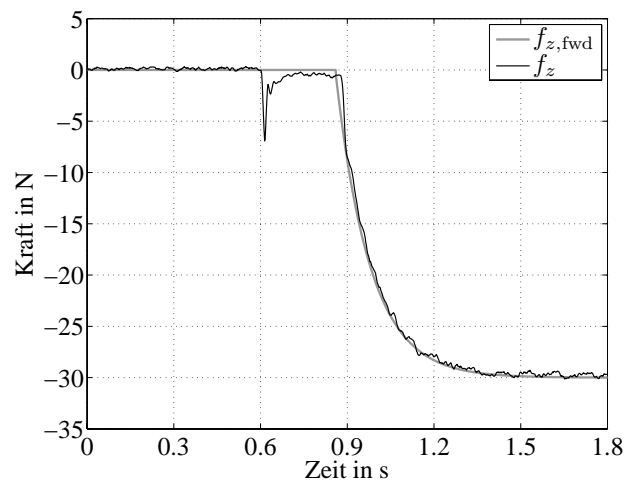
Die in diesem Versuchsaufbau an der künstlichen Umgebung eingestellte Umgebungssteifigkeit beträgt  $c_e = 25\text{ N/mm}$ . Damit die gewünschte Endeffektorkraft von  $f_z = -30\text{ N}$  entsteht, muss der Tauchspulenaktor  $\Delta z = 1,2\text{ mm}$  eintauchen. Die bei dieser Bewegung entstehende Reibung und das Messrauschen verursachen die erkennbaren Störungen der Endeffektorkraft.

Die oben beschriebene Funktionsweise gilt auch für die in Abbildung 6.4(c) gezeigte Momentenregelung um die  $z$ -Achse des Parallelroboters. Die Dämpfungsphase im Zeitraum  $t = [0,4\text{ s}; 0,65\text{ s}]$  ist deutlich zu erkennen. Da sich im Versuchsaufbau der Kontaktpunkt während der Momentenregelung verschiebt, ändert sich auch der effektive Hebelarm. Dies zeigt sich in den größeren Abweichungen des gemessenen Endeffektormoments von der Vorsteuerung.

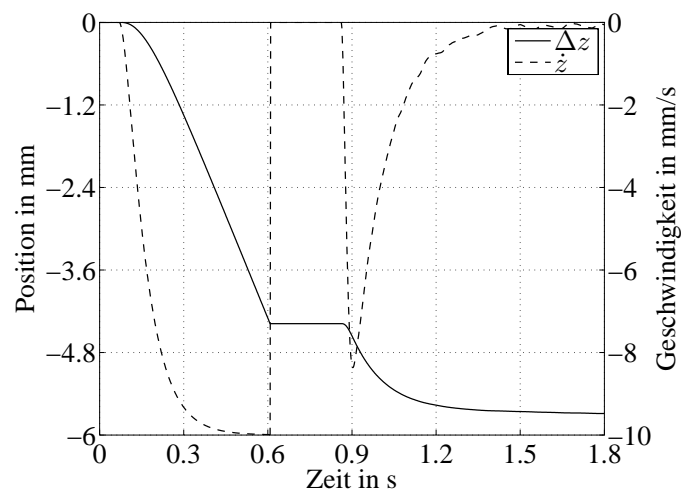
### 6.3.2 Vergleich der Kontaktregler mit und ohne Dämpfungsphase

Ein Vergleich der Kontaktregelung mit und ohne Dämpfungsphase verdeutlicht die Notwendigkeit der gewählten globalen Betrachtungsweise. Der Versuchsaufbau besteht bei diesem Experiment aus der Kunststoffumgebung, die eine Umgebungssteifigkeit von ungefähr  $c_{e,0} \approx 200\text{ N/mm}$  besitzt. Der Endeffektor des Parallelroboters mit dem gegriffenen Kunststoffklotz (siehe Abbildung 6.1(b)) soll eine Kraft von  $f_{z,\text{ref}} = -50\text{ N}$  aufbringen. Die Annäherungsgeschwindigkeit beträgt  $\dot{z} = -10\text{ mm/s}$ . Eine schnellere Annäherung, die ein Abprallen des Endeffektors verursachen würde, wie in Abschnitt 5.3.2 simuliert, kann mit dem Versuchsträger nicht durchgeführt werden, da sonst eine Beschädigung des Parallelroboters nicht auszuschließen wäre.

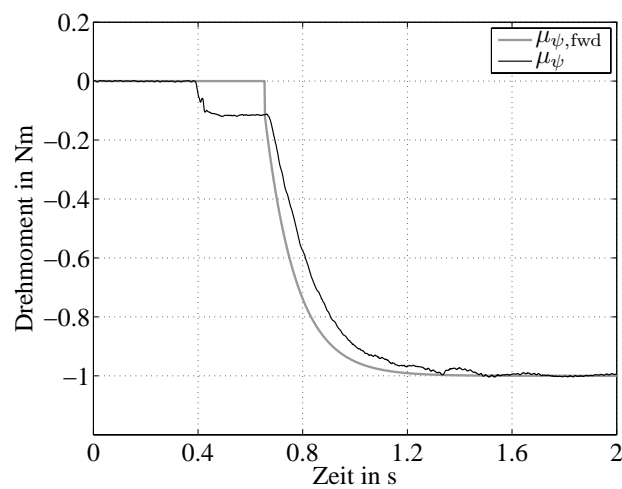
In Abbildung 6.5 ist ein Vergleich der Kontaktregelungen mit Dämpfungsphase bzw. mit



(a) Kraftregelung in der  $z$ -Achse des Roboterkoordinatensystems



(b) Solltrajektorie der Kraftregelung in der  $z$ -Achse des Handkoordinatensystems



(c) Momentenregelung um die  $z$ -Achse des Roboterkoordinatensystems

Abb. 6.4: Funktionsprinzip der Kontaktregelung mit Transitionsphase

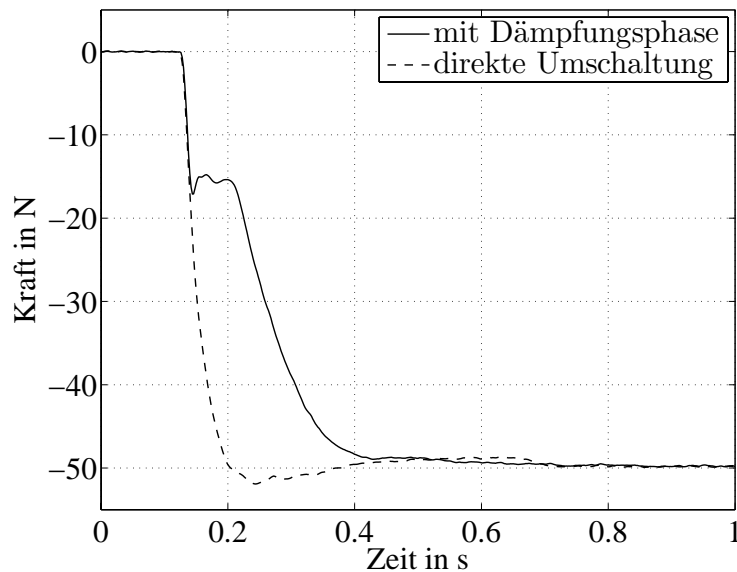


Abb. 6.5: Vergleich zwischen der Kraft-Momentenregelung mit direkter Umschaltung und mit Dämpfungsphase

direkter Umschaltung zu sehen. Im Vergleich zu der in Abschnitt 6.3.1 untersuchten Regelung ist die Phase auf  $t_{\text{damp}} = 75 \text{ ms}$  verkürzt worden. Diese Zeit wurde empirisch ermittelt und stellt einen guten Kompromiss zwischen ausreichender Dämpfung der Transitionseffekte und Geschwindigkeit der Kraft-Momentenregelung dar. Die direkte Kontaktregelung ohne Dämpfungsphase verursacht ein Überschwingen von  $\Delta \hat{f}_z = 4\%$ . Eine Erhöhung der Annäherungsgeschwindigkeit und eine größere Umgebungssteifigkeit führen zu ausgeprägteren Überschwingungen. Für die Montage ist daher der Einsatz einer Kraft-Momentenregelung mit Dämpfungsphase vorzuziehen.

### 6.3.3 Konturverfolgung

Die Möglichkeiten der praktischen Anwendung der Kontaktregelung werden im Rahmen einer hybriden Regelungsaufgabe untersucht. Die Abbildung 6.6 zeigt die Messungen, die während der Abarbeitung eines Aktionsprimitivnetzes aufgenommen wurden, das zur Verfolgung der in Abbildung 6.1(b) beschriebenen Kontur programmiert wurde. Der Endeffektor des Parallelroboters nimmt zuerst Kontakt mit der Umgebung auf und übt eine Kraft von  $f_{y,\text{ref}} = -10 \text{ N}$  aus. Ist diese Kontaktaufnahme abgeschlossen, bewegt sich der Endeffektor mit  $10 \text{ mm/s}$  entlang der  $x$ -Achse des Roboterkoordinatensystems im Arbeitsraum. Die Kraft in  $y$ -Richtung soll während der ganzen Zeit aufrecht erhalten werden, wodurch der Endeffektor den Ausformungen der Umgebung folgt (siehe Abbildung 6.6(a)). Der Kraftverlauf ist abhängig von der Umgebungskontur und in Abbildung 6.6(b) dargestellt. Für  $2,5 \text{ cm}$  verläuft die Umgebungsgrenze parallel zur  $x$ -Achse und die Endeffektorkraft ist  $f_y = -10 \text{ N}$ . Vorhandene Störungen sind auf die Reibung zurückzuführen. In den nächsten  $5 \text{ cm}$  folgt

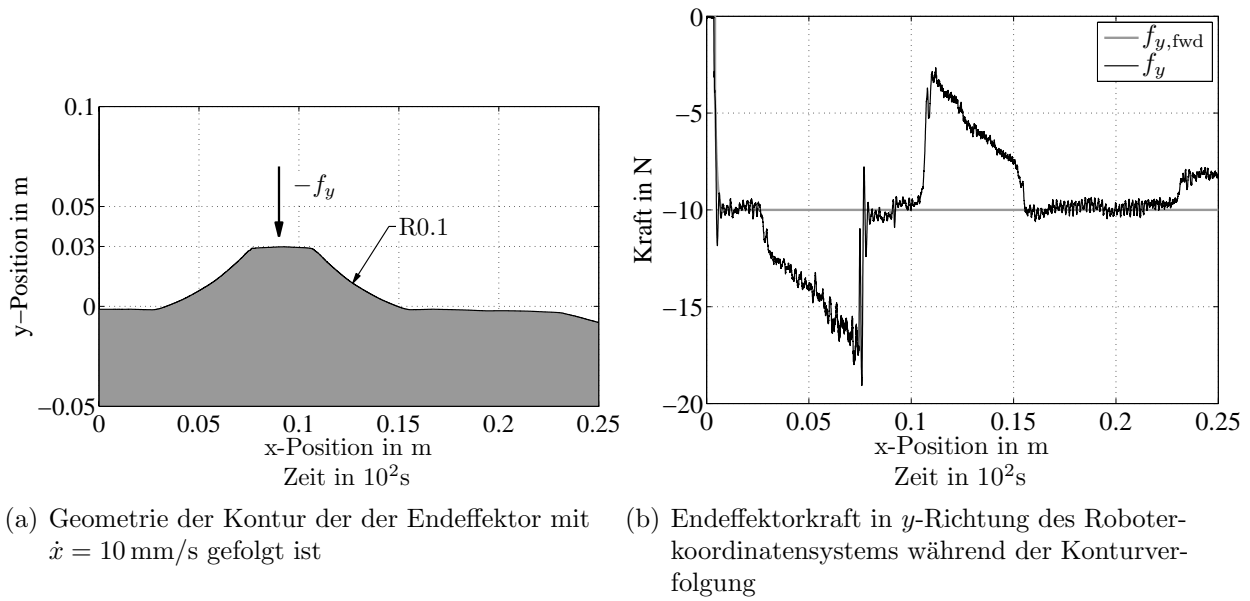


Abb. 6.6: Konturverfolgung mit hybrider Regelung

der Endeffektor einer kreisförmigen Umgebung mit Radius 10 cm. Die stetig anwachsende Steigung der Kontur bewirkt einen sich erhöhenden Regelfehler der Endeffektorkraft. Diese erreicht Spitzenwerte von fast  $f_y = -20 \text{ N}$ . Auf dem folgenden geraden Teilstück stellt sich sofort wieder die Sollkraft ein. Von 11 cm bis 15 cm nimmt die Steigung der Kontur in Wirkungsrichtung der Endeffektorkraft stetig ab. Analog zu dem obigen Verhalten verringert sich die Endeffektorkraft auf minimal  $f_y = -3 \text{ N}$ , wird aber nie null, d. h. der Endeffektor verliert nie den Kontakt zur Umgebung.

Das obige Beispiel zeigt die praktische Anwendbarkeit der vorgestellten Kontaktregelung im Rahmen einer hybriden Regelungsaufgabe. Die notwendige Entkoppelung der Freiheitsgrade des Parallelroboters in der unterlagerten Antriebsregelung ist gegeben. Darüber hinaus zeigt sich in diesem Experiment die Bedeutung des in Korollar 5 beschriebenen Einflusses der Umgebungsgeometrie und des Task-Frames auf den Kraftverlauf. Eine Anpassung der Orientierung des Task-Frames an die Kontur würde die Abweichungen der Endeffektorkraft reduzieren.

### 6.3.4 Orthogonale und verkoppelte Regelziele

Die Diskussion der in Abschnitt 5.4.1 und Korollar 6 formulierten Stabilitätsprobleme der Kontaktregelung bei *verkoppelten* Regelzielen soll in den beiden abschließenden Experimenten praktisch dargestellt werden. Beide in Abbildung 6.7 dargestellten Messungen zeigen den Kraftverlauf einer Kontaktregelung in zwei Koordinatenachsen des Task-Frames.

Im ersten Experiment soll der Endeffektor eine Kraft in die negative  $z$ -Richtung und gleichzeitig ein Moment um diese Koordinatenachse ausüben. Nach Gleichung (5.47) beeinflussen sich diese Regelungsziele nicht. Die durch das Moment induzierten Kräfte  $f_x$  und

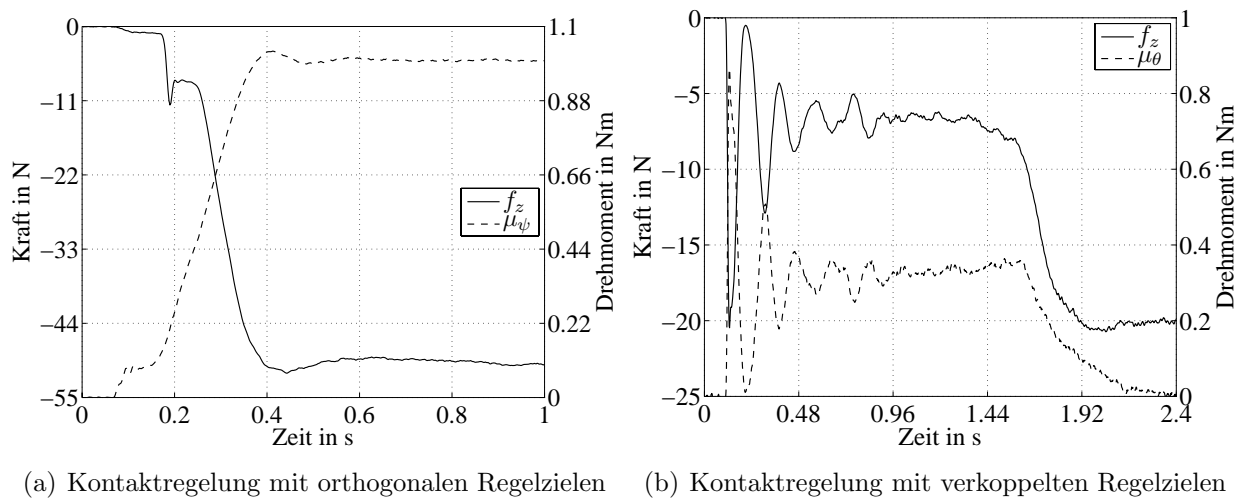


Abb. 6.7: Kontaktregelung in zwei Koordinatenachsen mit orthogonalen und verkoppelten Regelzielen

$f_y$  sind Störgrößen der in diesen Achsen im Eingriff befindlichen Lageregler. Die *orthogonalen* Regelziele der Kontaktregler beeinflussen sich nicht gegenseitig und ermöglichen ein störungsfreies Einschwingverhalten beider Regelgrößen, das in Abbildung 6.7(a) dargestellt ist.

Demgegenüber ist in Abbildung 6.7(b) deutlich die gegenseitige Beeinflussung der Kontaktregler in den Koordinatenachsen des Task-Frames zu sehen. Die Regelungsaufgabe besteht hier in der Ausrichtung eines beliebig um die  $y$ -Achse orientierten Endeffektors an einer ebenen Oberfläche in der  $x$ - $y$ -Ebene. Das Aktionsprimitiv ist so formuliert, dass der Endeffektor eine Kraft von  $f_{z,\text{ref}} = -20$  N auf die Oberfläche ausüben soll und das Drehmoment gleichzeitig zu  $\mu_{\theta,\text{ref}} = 0$  Nm geregelt wird. Nach Gleichung (5.47) beeinflusst die Kraft das zu regelnde Moment und umgekehrt. Beide Regelziele sind *verkoppelt*, was letztendlich auf einen verlängerten Einschwingvorgang mit deutlichem Überschwingen führt.





# Kapitel 7

## Zusammenfassung und Ausblick

IN der vorliegenden Arbeit wird ein Regelungskonzept zur Kontaktregelung von Parallelrobotern beschrieben, das die Programmierung von Montageaufgaben auf der Basis von Aktionsprimitiven als sensorbasierte und fehlertolerante Roboterprogrammiersprache ermöglicht.

Die besonderen Eigenschaften von Parallelrobotern werden durch die unterlagerte Antriebsregelung berücksichtigt. Die Auslegung als Kaskadenregelung mit innerem Geschwindigkeitsregelkreis und übergeordneter Lageregelung in Gelenkkoordinaten verwendet das analytisch lösbare inverse kinematische Problem. Verzögerungen durch die Umrichter und Messwertfilter werden in der Reglerauslegung berücksichtigt. Die Integration des inversen dynamischen Modells ermöglicht die Berechnung geeigneter Vorsteuergrößen und damit die Entkoppelung der einzelnen Freiheitsgrade des Parallelroboters. Der eingeführte Stabilitätsbeweis der unterlagerten Antriebsregelung mit Hilfe der Passivitätstheorie berücksichtigt den Einfluss der Coriolis- und Zentrifugalkräfte sowie die Lageabhängigkeit der Massenträgheitsmatrix. Die unterlagerte Antriebsregelung kapselt den nichtlinearen Parallelroboter und garantiert in dem gesamten Arbeitsraum die Einhaltung der dynamischen Parameter. Desweiteren stellt sie eine kartesische Schnittstelle für die Bewegungsmodule und den Task-Frame-Formalismus zur Verfügung und erfüllt so alle Voraussetzungen für den Einsatz in einer Steuerungsarchitektur zur Verarbeitung von Aktionsprimitiven.

Die Entwicklung einer Beschreibung der Systemdynamik im Umgebungskontakt durch die Definition eines Kontaktkoordinatensystems berücksichtigt erstmalig die Anforderungen des 'Task-Frame-Formalismus'. Dadurch ist u. a. die Ableitung von Bedingungen zur Aufgabenplanung mit Aktionsprimitivnetzen möglich, die in zukünftigen Arbeiten zur Montageplanung einfließen können.

Die umfassende Betrachtung der Kontaktregelung mit den drei Phasen der Freiraumbewegung, der Transitionsphase und des Kontakts führt auf eine struktumschaltende Regelung. Sie ist als Bewegungsmodul der Steuerungsarchitektur realisiert, die einen weiteren Regelkreis um die unterlagerte Antriebsregelung schließt und umfasst einen Transitionsplaner sowie einen Kraft-Momentenregler.

Besondere Aufmerksamkeit wird dem Verhalten des Parallelroboters in der Transitionsphase gewidmet. Die Beschreibung des Systemverhaltens durch die Maßtheorie und insbesondere durch Moreaus Stoßprozess beachtet die gültigen Gesetzmäßigkeiten der nicht-glatte

Mechanik. Zugleich stellen sie die notwendigen Werkzeuge zur numerischen Simulation unilateraler und multilateraler Umgebungskontakte zur Verfügung, die zur Validierung der Transitionsplaner eingesetzt werden. Die Untersuchungen des unilateralen Umgebungskontakts im Rahmen dieser Arbeit können in weiteren Projekten mit Hilfe dieser Methode auf multilaterale Kontaktsituationen erweitert werden.

Eine Anpassung des Führungsgrößengenerators an das Verhalten der unterlagerten Antriebsregelung erlaubt die Interpretation der Kraft-Momentenregelung als Modellfolgeregelung mit der Möglichkeit einer analytisch geschlossenen Reglerauslegung. Praktische Versuche zeigen die Eignung des vorgestellten Konzepts für die Montage mit Parallelrobotern.

Die Erfahrungen während der Arbeit an dieser Dissertation zeigen eine Möglichkeit zur Verbesserung der Kontaktregelung auf. Eine Erweiterung der Steuerungsarchitektur ermöglicht es über die Kraft-Momentenvorsteuerung direkten Zugriff auf die Drehmomentschnittstelle des Parallelroboter zu erlangen. Dadurch sind Regelungskonzepte zu realisieren, die über die Vorgabe einer Lageinformation alleine nicht möglich sind. Die aktive Dämpfung von Abpralleffekten in der Transitionsphase kann nur sinnvoll mit einem Stelleingriff in dieser Ebene angegangen werden. Bleibt die Einflussmöglichkeit der Vorsteuerung nicht auf die kraft-momentengeregelten Freiheitsgrade beschränkt, ist außerdem eine Kompensation der Reibung in den freien Bewegungsrichtungen des Endeffektors realisierbar.

Die zusätzliche Integration des elastischen Robotermodells, wie sie in der Auslegung des Kraft-Momentenreglers für das dynamische Umgebungsmodell erwähnt wurde, berücksichtigt die Verformungen der Struktur im Umgebungskontakt und verbessert so die Entkopplung von beschränkten und freien Koordinatenachsen des Task-Frames.

# Anhang A

## Roboter

In diesem Kapitel des Anhangs befinden sich die Beschreibungen der Roboter HEXA und COMAU SMART, die in den Beispielrechnungen, Simulationen und Experimenten der obigen Kapitel zitiert wurden.

### A.1 Parallelroboter HEXA

Der Parallelroboter HEXA (siehe Abbildung A.1(a)) wurde im Rahmen des Sonderforschungsbereichs 562 – *Parallelroboter für Handhabung und Montage* aufgebaut und besitzt sechs Freiheitsgrade. Die verwendeten Direktantriebe sind in drei Paaren mit kollinearen Achsen in einer Ebene angebracht. Von jedem Antrieb führt eine kinematische Kette, bestehend aus einer Kurbel mit der Länge 0,24 m und einem Stab der Länge 0,564 m zu der Arbeitsplattform. Der nutzbare Arbeitsraum hat einen maximalen Durchmesser von ca. 0,5 m, eine Höhe von 0,2 m und befindet sich unterhalb der Ebene in der die Antriebe montiert sind.

Die Leistung der Antriebe und die Struktur ist, bei einer Nutzlast von 4 kg, für eine maximale Beschleunigung von  $50 \text{ m/s}^2$  und eine Geschwindigkeit von 5 m/s ausgelegt. Die Masse der Struktur ohne Nutzlast beträgt 15 kg, wovon 1,6 kg auf die Arbeitsplattform entfallen. Für praktische Versuche ist zusätzlich zu dem obligatorischen Greifer und Kraft-Momentensensor ein Kollisionsschutz montiert. Die Masse dieser Geräte beträgt insgesamt ca. 1 kg.

Von Rose [2004] ist die dynamische Modellierung der Parallelstruktur nach dem Jourdainischen Prinzip beschrieben. Die Implementierung des inversen dynamischen Modells in kartesischen Koordinaten erfolgte in der Programmiersprache C. Zusätzlich ist eine Schnittstelle zu dem Programmpaket *Matlab/Simulink*<sup>1</sup> realisiert, die eine Simulation des Parallelroboters ermöglicht.

---

<sup>1</sup>Matlab und Simulink sind eingetragene Warenzeichen der Firma The MathWorks, Inc.



(a) Parallelroboter HEXA



(b) Serieller Roboter COMAU SMART-3 S

Abb. A.1: Roboter

## A.2 Serieller Roboter COMAU SMART

Der serieller Industrieroboter COMAU SMART-3 S (siehe Abbildung A.1(b)) der Firma Comau S.p.A. mit sechs Freiheitsgraden dient in Simulationen dem Vergleich zwischen den verschiedenen Robotertypen. Die Struktur besteht aus einer offenen kinematischen Kette mit sechs rotatorischen Antrieben, deren Verbindungsglieder eine Länge zwischen 0,1 m und 0,6 m besitzen. Diese führt auf einen annähernd kugelförmigen Arbeitsraum mit einem Radius von ca. 1,4 m. Die maximale Nutzlast am Endeffektor beträgt 6 kg bei einer Entfernung von 0,215 m vom Zentrum des Arbeitsraums.

Eine Modellierung des Roboters in Siciliano und Villani [1999] beschreibt das direkte kinematische Problem und die differentielle Kinematik. Das dynamische Modell ist in Gelenkkoordinaten angegeben und berücksichtigt zusätzlich die Reibung in den Getrieben. Aus diesen Angaben wurde im Rahmen der begleitenden Arbeiten dieser Dissertation eine Simulationsumgebung für *Matlab/Simulink* aufgebaut.

# Anhang B

## Passivitätstheorie

Dieses Kapitel des Anhangs gibt einen stichwortartigen Überblick des Konzepts der Passivität und dessen Anwendung zum Stabilitätsbeweis linearer und nichtlinearer Systeme.

### B.1 Einführung und Definitionen

Passivität ist eine Eigenschaft, die für lineare und nichtlineare Systeme der Form

$$\Sigma := \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}) \end{cases} \quad (\text{B.1})$$

nachgewiesen werden kann. In obiger Gleichung ist  $\mathbf{x} \in \mathbb{R}^n$  der Zustand des Systems,  $\mathbf{u}$  dessen Eingang und  $\mathbf{y}$  der Ausgang. Passivität baut auf der allgemeineren Definition über dissipative Systeme auf und wird von Ortega u. a. [1998, Anhang A] sowie Khalil [2002] ausführlich beschrieben. Für beide Eigenschaften müssen zwei Funktionen eingeführt werden:

**Definition 6 (Versorgungsrate)** Die *Versorgungsrate* (engl. supply rate) definiert die Energie, die dem System (B.1) zugeführt wird. Es sei  $\mathbf{u} \in \mathbb{R}^m$  der Eingang des Systems und  $\mathbf{y} \in \mathbb{R}^m$  dessen Ausgang. Die Ordnung beider Signale ist  $m$ . Dann ist

$$w(\mathbf{u}, \mathbf{y}) : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R} \quad (\text{B.2})$$

die Versorgungsrate des Systems.

**Definition 7 (Speicherfunktion)** Die Speicherfunktion (engl. storage function)  $\mathcal{H} : \mathbb{R}^n \rightarrow \mathbb{R}_{0+}$  repräsentiert die gesamte im System (B.1) gespeicherte Energie:

$$\mathcal{H}(\mathbf{x}(t)) = \mathcal{T}(\mathbf{x}(t)) + \mathcal{V}(\mathbf{x}(t)) = \text{kinetische Energie} + \text{potentielle Energie} \quad (\text{B.3})$$

Beide Funktionen sind über die Dissipativitätsungleichung miteinander verknüpft, die aussagt, dass:

**Definition 8 (Dissipativität)**

$$\mathcal{H}(\mathbf{x}(T)) \leq \mathcal{H}(\mathbf{x}(0)) + \int_0^T w(\mathbf{u}(t), \mathbf{y}(t)) dt \quad (\text{B.4})$$

- (a) Entlang den Trajektorien eines dissipativen Systems ist die Versorgungsrate nicht kleiner als die Zunahme der Speicherfunktion.
- (b) Ein dissipatives System kann nicht mehr Energie speichern als ihm von außen zugeführt wird. Die verbleibende Differenz wird dissipative Energie genannt.

In der differentiellen Form lautet die Dissipativitätsungleichung

$$\dot{\mathcal{H}}(\mathbf{x}(t)) \leq w(\mathbf{u}(t), \mathbf{y}(t)). \quad (\text{B.5})$$

Die Eigenschaft der Passivität erlaubt eine weitergehende Aussage über das Systemverhalten: Durch die folgende Definition wird die Eigenschaft der Versorgungsrate dissipativer Systeme festgelegt.

**Definition 9 (Passivität)** Ein passives System ist definiert durch:

- (a) Ein System (B.1) ist *passiv*, wenn es dissipativ mit der Versorgungsrate  $w(\mathbf{u}, \mathbf{y}) = \mathbf{u}^T \mathbf{y}$  ist:

$$\dot{\mathcal{H}}(\mathbf{x}(t)) \leq \mathbf{u}^T(t) \mathbf{y}(t) \quad \text{bzw.} \quad (\text{B.6})$$

$$\mathcal{H}(\mathbf{x}(T)) \leq \mathcal{H}(\mathbf{x}(0)) + \int_0^T \mathbf{u}^T(t), \mathbf{y}(t) dt. \quad (\text{B.7})$$

Darüber hinaus gelten die weiteren Spezialisierungen:

- (a) Ist System (B.1) bezüglich der Versorgungsrate  $w(\mathbf{u}, \mathbf{y}) = \mathbf{u}^T \mathbf{y} - \alpha \|\mathbf{u}\|^2$  mit  $\alpha \in \mathbb{R}$  dissipativ, so ist es  $\alpha$ -*eingangspassiv*.
- (b) Ist das System (B.1) bezüglich der Versorgungsrate  $w(\mathbf{u}, \mathbf{y}) = \mathbf{u}^T \mathbf{y} - \beta \|\mathbf{y}\|^2$  mit  $\beta \in \mathbb{R}$  dissipativ, so ist es  $\beta$ -*ausgangspassiv*.

## B.2 Zusammenschaltung passiver Systeme

Die Passivitätseigenschaften einzelner Systeme bleiben bei verschiedenen Formen der Zusammenschaltung unter bestimmten Bedingungen bestehen. Die Parallelschaltung und Rückkopplung passiver Systeme, die in Abbildung B.1(a) bzw. B.1(b) gezeigt wird, ist ebenfalls

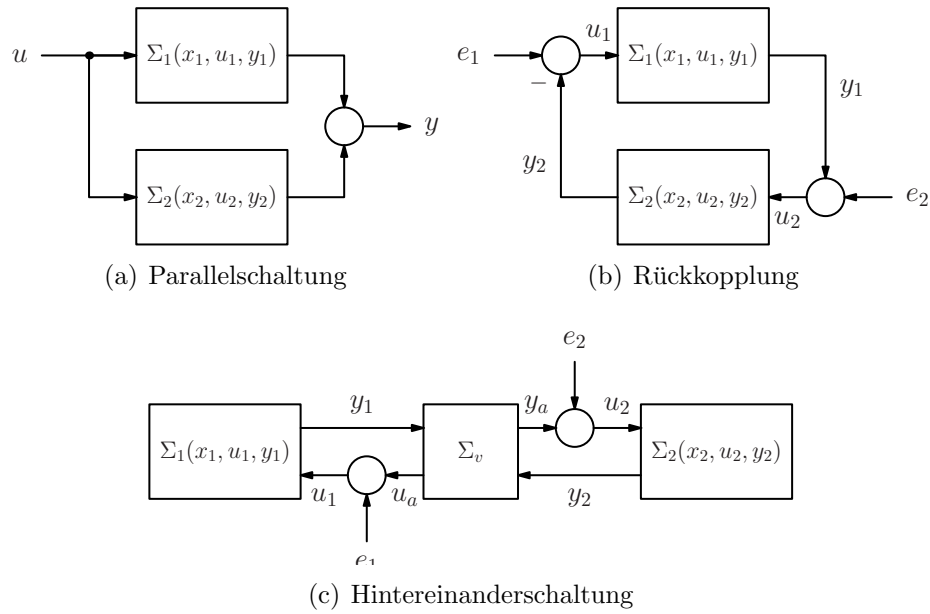


Abb. B.1: Zusammenschaltung passiver Systeme

passiv. Gleiches gilt für die Hintereinanderschaltung aus Abbildung B.1(c), wenn das Verbindungssystem  $\Sigma_v$  verlustlos ist, d. h. die Gleichung

$$\int_0^T (\mathbf{y}_1 \mathbf{u}_a^T + \mathbf{y}_2 \mathbf{y}_a^T) dt = 0 \quad (\text{B.8})$$

erfüllt wird. Dann ist die Speicherfunktion  $\mathcal{H} = \mathcal{H}_1 + \mathcal{H}_2$  und das Gesamtsystem vom Eingang  $u$  bzw.  $(e_1, e_2)$  zum Ausgang  $y$  bzw.  $(y_1, y_2)$  passiv.

### B.3 Passivität von LTI-Systemen

Das Konzept der Passivität lässt sich auch auf LTI-Systeme anwenden. In diesem Fall ist die Eigenschaft der positiven Reellheit [vgl. Slotine und Li, 1991] eng mit der Passivität verknüpft.

**Definition 10 (Positiv reelle Übertragungsfunktion)** Eine eindimensionale, rationale Übertragungsfunktion  $G(s)$  ist positiv reell, wenn

- alle Pole  $s_i$  von  $G(s)$  einen Realteil kleiner oder gleich null haben,

$$\operatorname{Re}\{s_i\} \leq 0 \quad (\text{B.9})$$

- die Ortskurve für alle  $\omega \in \mathbb{R}$  in der geschlossenen rechten Halbebene liegt

$$\operatorname{Re}\{G(j\omega)\} \geq 0 \quad (\text{B.10})$$

und

- jede rein imaginäre Polstelle  $j\omega_0$  von  $G(s)$  einen einfachen Pol darstellt, dessen Residuum

$$\lim_{s \rightarrow j\omega_0} (s - j\omega_0)G(s) \quad (\text{B.11})$$

positiv und reell ist.

Sind alle Bedingungen der obigen Definition erfüllt, resultiert aus dem folgenden Satz die Passivität des LTI-Systems:

**Satz 1** *Jede lineare zeitinvariante und minimale Realisierung des Systems*

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (\text{B.12a})$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (\text{B.12b})$$

mit der Übertragungsfunktion  $\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$  ist passiv, wenn  $\mathbf{G}(s)$  positiv reell ist. Dabei ist  $\mathbf{A}$  die Systemmatrix,  $\mathbf{B}$  die Eingangsmatrix,  $\mathbf{C}$  die Ausgangsmatrix und  $\mathbf{D}$  die Durchgangsmatrix.

Die Speicherfunktion des LTI-Systems wird mit dem *Kalman-Yakubovich-Popov*-Lemma bestimmt.

**Lemma 2 (Kalman-Yakubovich-Popov)** *Es existiert eine positiv definite Matrix  $\mathbf{P}$  und eine positiv semidefinite Matrix  $\mathbf{Q}$  für die gilt*

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{Q} \quad (\text{B.13a})$$

$$\mathbf{P}\mathbf{B} = \mathbf{C}^T \quad (\text{B.13b})$$

dann lautet die Speicherfunktion  $\mathcal{H} = \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x}$ .

## B.4 Passivität von Euler-Lagrange-Systemen

Die Bewegungsgleichungen der in dieser Arbeit untersuchten Parallelroboter gehören zu der Klasse der Euler-Lagrange-Systeme. Für diese ist nach Ortega u. a. [1998] und Arimoto [2000] ein passiver Operator vom Eingang  $\mathbf{u}$  zu den *Geschwindigkeiten*  $\dot{\mathbf{q}}$

$$\Sigma : \mathbf{u} \mapsto \dot{\mathbf{q}} \quad (\text{B.14})$$

mit der Speicherfunktion

$$\mathcal{H} = \frac{1}{2}\dot{\mathbf{q}}^T\mathbf{M}\dot{\mathbf{q}} + \mathcal{V} \quad (\text{B.15})$$



definiert. Dabei ist  $\mathbf{M}$  die Massenträgheitsmatrix und  $\mathcal{V}_g$  die potentielle Energie. Dadurch sind auch Euler-Lagrange-Systeme für Stabilitätsuntersuchungen auf der Basis der Passivitätstheorie zugänglich.



# Literaturverzeichnis

Die Literaturangaben sind alphabetisch nach den Namen der Autoren und nicht nach dem Auftreten im Dokument sortiert. Bei mehreren Autoren wird nach dem ersten Autor sortiert.

- [Acary und Brogliato 2005] ACARY, V. ; BROGLIATO, B.: Numerical Time Integration of Higher Order Dynamical Systems with State Constraints. In: *Proc. of. 5th EUROMECH Nonlinear Dynamics Conf. 2005 (ENOC 2005)*. Eindhoven, Netherlands, August 2005
- [Arimoto 2000] ARIMOTO, S.: Passivity-Based Control. In: *Proc. IEEE Int. Conf. on Robotics & Automation (ICRA 2000)*. San Francisco, USA, April 2000, S. 227–232
- [Assuncao und Schumacher 2003] ASSUNCAO, V. ; SCHUMACHER, W.: Hybrid Force Control for Parallel Manipulators. In: *11th Mediterranean Conference on Control and Automation*. Rhodes, Greece, Juni 2003
- [Brogliato 1999] BROGLIATO, B.: *Nonsmooth Mechanics*. 2. Auflage. London, UK : Springer-Verlag, 1999
- [Brunotte 2001] BRUNOTTE, Ch.: *Regelung und Identifikation von Linearmotoren für Werkzeugmaschinen*, Technische Universität Braunschweig, Dissertation, 2001
- [Bruyninckx und De Schutter 1996] BRUYNINCKX, H. ; DE SCHUTTER, J.: Specification of Force-Controlled Actions in the Task Frame Formalism – A Synthesis. In: *IEEE Trans. Robot. Automat.* 12 (1996), August, Nr. 4, S. 581–589
- [Canudas de Wit u. a. 1996] CANUDAS DE WIT, C. (Hrsg.) ; SICILIANO, B. (Hrsg.) ; BASTIN, G. (Hrsg.): *Theory of Robot Control*. Kap. 4.3 Hybrid force/motion control, S. 156–169. London, UK : Springer-Verlag, 1996 (Communications and Control Engineering Series)
- [Cheah u. a. 2003] CHEAH, C.C. ; KAWAMURA, S. ; ARIMOTO, S.: Stability of hybrid position and force control for robotic manipulator with kinematics and dynamics uncertainties. In: *Automatica* 39 (2003), S. 847–855
- [Chiaverini und Sciavicco 1993] CHIAVERINI, S. ; SCIAVICCO, L.: The Parallel Approach to Force/Position Control of Robotic Manipulators. In: *IEEE Trans. Robot. Automat.* 9 (1993), August, S. 361–373

- [FANUC Robotics 2005] FANUC ROBOTICS: *Intelligent Robots with FANUC Force Sensors*. 2005
- [Finkemeyer 2004] FINKEMEYER, B.: *Robotersteuerungsarchitektur auf der Basis von Aktionsprimitiven*, Technische Universität Braunschweig, Dissertation, 2004
- [Finkemeyer u. a. 2005a] FINKEMEYER, B. ; KRÖGER, T. ; WAHL, F. M.: Aktionsprimitive: Ein universelles Roboter-Programmierparadigma. In: *at – Automatisierungstechnik* 53 (2005), April, Nr. 4, S. 189–196
- [Finkemeyer u. a. 2005b] FINKEMEYER, B. ; KRÖGER, T. ; WAHL, F. M.: Executing Assembly Tasks Specified by Manipulation Primitive Nets. In: *Advanced Robotics, VSP and Robotics Society of Japan* 19 (2005), Juni, S. 591–611
- [Föllinger 1990] FÖLLINGER, O.: *Regelungstechnik*. Hüthig Verlag, 1990
- [Grundler 2004] GRUNDLER, E.: Automatica 2004 – Roboter im Vordergrund. In: *VDI-Z Integrierte Produktion* 146 (2004), Nr. 9, S. 60
- [Hasegawa u. a. 1992] HASEGAWA, T. ; SUEHIRO, T. ; TAKASE, K.: A model-based manipulation system with skill-based execution. In: *IEEE Trans. Robot. Automat.* 8 (1992), Oktober, Nr. 5, S. 535 – 544
- [Hesselbach u. a. 2004] HESSELBACH, J. ; BIER, C. ; PIETSCH, I. ; PLITEA, N. ; BÜTTGENBACH, S. ; WOGERSIEN, A. ; GÜTTLER, J.: Passive joint-sensor applications for parallel robots. In: *Proc. of. 2004 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS 2004)* 4 (2004), S. 3507–3512
- [Hesselbach u. a. 2005] HESSELBACH, J. ; MAASS, J. ; BIER, C.: Singularity Prediction for Parallel Robots for Improvment of Sensor-Integrated Assembly. In: *Annals of the CIRP* 54 (2005), S. 349–352
- [Hogan 1985] HOGAN, N.: Impedance Control: An Approach to Manipulation: Part I – Theory, Part II – Implementation, Part III – Applications. In: *ASME Journal of Dynamic Systems, Measurement, and Control* 107 (1985), S. 1–24
- [Karllson 2005] KARLLSON, J. (Hrsg.): *World Robotics 2005 – Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment*. Genf, Schweiz : United Nations Economic Commission for Europe (UNECE) und International Federation of Robotics (IFR), 2005. – URL [http://www.ifr.org/publications/World\\_Robotics.htm](http://www.ifr.org/publications/World_Robotics.htm)
- [Kautsky und Nichols 1985] KAUTSKY, J. ; NICHOLS, N. K.: Robust Pole Assignment in Linear State Feedback. In: *Int. J. Control* 41 (1985), S. 1129–1155
- [Khalil 2002] KHALIL, H. K.: *Nonlinear Systems*. 3. Auflage. Prentice-Hall, 2002

- [Kock 2001] KOCK, S.: *Parallelroboter mit Antriebsredundanz*. Düsseldorf, Germany : VDI-Verlag, 2001 (Fortschrittsberichte VDI-Reihe 8, Nr. 890)
- [Kohn u. a. 2004] KOHN, N. ; STEINER, J. ; VARCHMIN, J.-U. ; GOLTZ, U.: Universal Communication Architecture for High-Dynamic Robot Systems using QNX. In: *Proc. of 8. Intl. Conf. on Control, Automation, Robotics and Vision (ICARCV 04)*. Kunming, China, 2004
- [Kolbus u. a. 2005] KOLBUS, M. ; MAASS, J. ; REISINGER, T.: Robot Control based on Skill Primitives. In: *Proc. of IASTED Conf. on Robotics & Applications 2005*. Cambridge, USA, 2005
- [Kröger u. a. 2004] KRÖGER, T. ; FINKEMEYER, B. ; WAHL, F. M.: A Task Frame Formalism for Practical Implementations. In: *Proc. IEEE Int Conf. on Robotics & Automation (ICRA 2004)*. New Orleans, USA, 2004, S. 5218–5223
- [Kugi 2005] KUGI, A.: Passivitäts- und Lyapunovbasierte Reglerentwurfsmethoden. In: *at – Automatisierungstechnik* 58 (2005), August, Nr. 8, S. 353–355. – Editorial
- [Kugi und Schlacher 2002] KUGI, A. ; SCHLACHER, K.: Analyse und Synthese nichtlinearer dissipativer Systeme: Ein Überblick. In: *at – Automatisierungstechnik* 50 (2002), Februar, Nr. 2, S. 63–69. – Zweiteiliger Aufsatz. Teil zwei folgte in *at* 3/2002, S. 103ff
- [Kunze und Monteiro Marques 2000] KUNZE, M. ; MONTEIRO MARQUES, M. D. P.: An Introduction to Moreau’s Sweeping Process. In: BROGLIATO, B. (Hrsg.): *Impacts in mechanical systems: analysis and modelling* Euromech (Veranst.), Springer-Verlag, Juni 2000 (Lecture notes in physics), S. 1–60
- [Leonhard 1990] LEONHARD, W.: Trajectory Control of a Multi-axes Robot with Electrical Servo Drives. In: *IEEE Control Syst. Mag.* 10 (1990), October, Nr. 6, S. 3–9. – Presented at the 1989 IEEE Conference on Industrial Electronics, Philadelphia, USA, November 6-10, 1989.
- [Leonhard 2000] LEONHARD, W.: *Regelung elektrischer Antriebe*. 2. Auflage. Berlin : Springer-Verlag, 2000
- [Loría 2004] LORÍA, A.: *Cascaded nonlinear time-varying systems: analysis and design*. <http://public.lss.supelec.fr/perso/loria/Teaching/course-casc-ipn.html>. März 2004. – Lecture notes from FAP 2004
- [Maaß u. a. 2006] MAASS, J. ; REISINGER, T. ; HESSELBACH, J. ; SCHUMACHER, W.: A Versatile Modular Control Architecture for Sensor Integrated Assembly. In: *Proc. Conf. ISR/Robotik 2006*. München, Mai 2006

- [Mason 1981] MASON, M. T.: Compliance and Force Control for Computer Controlled Manipulators. In: *IEEE Trans. Syst., Man, Cybern.* 6 (1981), Juni, Nr. 11, S. 418–432
- [McClamroch und Wang 1988] McCLAMROCH, N. H. ; WANG, D.: Feedback Stabilization and Tracking of Constrained Robots. In: *IEEE Trans. Automat. Contr.* 33 (1988), Mai, Nr. 5, S. 419–426
- [Merlet 2000] MERLET, J.-P.: *Parallel Robots*. Dordrecht, The Netherlands : Kluwer Academic Publishers, 2000
- [Mills und Goldenberg 1989] MILLS, J. K. ; GOLDENBERG, A. A.: Force and Position Control of Manipulators During Constrained Motion Tasks. In: *IEEE Trans. Robot. Automat.* 5 (1989), Februar, Nr. 1, S. 30–46
- [Mills und Lokhorst 1993] MILLS, J. K. ; LOKHORST, D. M.: Stability and Control of Robotic Manipulators During Contact/Noncontact Task Transition. In: *IEEE Trans. Robot. Automat.* 9 (1993), Juni, Nr. 3, S. 335–345
- [Müller 1993] MÜLLER, M.: *Roboter mit Tastsinn*, Technische Universität Braunschweig, Dissertation, 1993
- [Olomski 1989] OLOMSKI, J.: *Bahnplanung und Bahnführung von Industrierobotern*. Vieweg-Verlag, 1989 (Fortschritte der Robotik)
- [Ortega u. a. 1998] ORTEGA, R. ; LORÍA, A. ; NICKLASSON, P. J. ; SIRA-RAMÍREZ, H.: *Passivity-based Control of Euler-Lagrange Systems: Mechanical, Electrical and Electromechanical Applications*. Springer-Verlag, 1998 (Communications and Control Engineering)
- [Ott u. a. 2005] OTT, C. ; ALBU-SCHÄFFER, A. ; KUGI, A. ; STRAMIGIOLI, S. ; HIRZINGER, G.: Kartesische Impedanzregelung von Robotern mit elastischen Gelenken: Ein passivitätsbasierter Ansatz. In: *at – Automatisierungstechnik* 53 (2005), August, Nr. 8, S. 378–388
- [Oya u. a. 2003] OYA, M. ; SU, C.-Y. ; KATO, R.: Robust Adaptive Motion/Force Tracking Control of Uncertain Nonholonomic Mechanical Systems. In: *IEEE Trans. Robot. Automat.* 19 (2003), Nr. 1, S. 175–181
- [Pagilla 2001] PAGILLA, P. R.: Control of Contact Problems in Constrained Euler-Lagrange Systems. In: *IEEE Trans. Automat. Contr.* 46 (2001), Oktober, S. 1595–1598
- [Pagilla und Biao 2001] PAGILLA, R.P. ; BIAO, Y.: A Stable Transition Controller for Constrained Robots. In: *IEEE/ASME Trans. Mechatron.* 6 (2001), März, Nr. 1, S. 65–74

- [Panteley und Ortega 1997] PANTELEY, E. ; ORTEGA, R.: Cascaded control of feedback interconnected systems: Application to robots with AC drives. In: *Automatica* 33 (1997), Nr. 11, S. 1935–1947
- [Papiernik 1986] PAPIERNIK, W.: Betragsoptimum und Riccati-Regler. In: *at – Automatisierungstechnik* 34 (1986), Mai, Nr. 5, S. 201–207
- [Pietsch 2003] PIETSCH, I.: *Adaptive Steuerung und Regelung ebener Parallelroboter*, Technische Universität Braunschweig, Dissertation, 2003
- [Raibert und Craig 1981] RAIBERT, M. ; CRAIG, J. J.: Hybrid Position/Force Control of Manipulators. In: *ASME J. of Dyn. Systems, Measurement and Control* 120 (1981), Nr. 2, S. 126–133
- [Reisinger 2005] REISINGER, T.: Hybrid Position/Force-Control for Implementing Skill Primitives. In: *Proc. of ICGST Conf. on Automatic Control and Systems*. Kairo, Ägypten, Dezember 2005
- [Reisinger u. a. 2005] REISINGER, T. ; WOBBE, F. ; KOLBUS, M.: Hybrid Position/Force-Control of a Planar Parallel Robot. In: *Proc. of Conf. on Problems of Automated Electrodrives*. Alushta, Ukraine, 2005, S. 250–254
- [Rose 2004] ROSE, M.: Dynamische Modellierung der HEXA-Parallelstruktur / DLR Braunschweig, Institut für Strukturmechanik. März 2004 (IB 131-2004/13). – Forschungsbericht. Interner Bericht
- [Rothfuß u. a. 1997] ROTHFUSS, R. ; RUDOLPH, J. ; ZEITZ, M.: Flachheit: Ein neuer Zugang zur Steuerung und Regelung nichtlinearer Systeme. In: *at - Automatisierungstechnik* 45 (1997), November, Nr. 11, S. 517–525
- [Roy und Whitcomb 2002] ROY, J. ; WHITCOMB, L.: Adaptive Force Control of Position/Velocity Controlled Robots: Theory and Experiment. In: *IEEE Trans. Robot. Automat.* 18 (2002), April, Nr. 2, S. 121–137
- [Sciavicco und Siciliano 2000] SCIAVICCO, L. ; SICILIANO, B.: *Modelling and Control of Robot Manipulators*. London, UK : Springer-Verlag, 2000
- [Sepulchre u. a. 1997] SEPULCHRE, R. ; JANKOVIĆ, M. ; KOKOTOVIĆ, P. V.: *Constructive Nonlinear Control*. Springer-Verlag, 1997 (Communications and Control Engineering)
- [Seul Jung u. a. 2004] SEUL JUNG ; HSIA, T.C. ; BONITZ, R.G.: Force Tracking impedance Control of Robot Manipulators Under Unknown Environment. In: *IEEE Trans. Contr. Sys. Tech.* 12 (2004), Mai, Nr. 3, S. 474–483

- [Siciliano und Villani 1999] SICILIANO, B. ; VILLANI, L.: *Robot Force Control*. Dordrecht, The Netherlands : Kluwer Academic Publisher, 1999
- [Slotine und Asada 1986] SLOTINE, J.-J. E. ; ASADA, H.: *Robot Analysis and Control*. New York, USA : John Wiley and Sons, 1986
- [Slotine und Li 1991] SLOTINE, J.-J. E. ; LI, W.: *Applied Nonlinear Control*. London, UK : Prentice-Hall, 1991
- [Thomas u. a. 2004] THOMAS, U. ; FLÖRKE, J. ; DETERING, S. ; WAHL, F. M.: An Integrative Approach for Multi-Sensor based Robot Task Programming. In: *Proc. of IEEE Intl. Conf. on Robotics & Automation (ICRA 2004)*. New Orleans, USA, 2004, S. 1149–1154
- [Thomas u. a. 2005] THOMAS, U. ; MAASS, J. ; WAHL, F. M. ; HESSELBACH, J.: Towards a New Concept of Robot Programming in High Speed Assembly Applications. In: *Proc. of IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS 2005)*. Kanada, 2005, S. 3932–3928
- [Tornambe 1999] TORNAMBE, A.: Modeling and Control of Impact in Mechanical Systems: Theory and Experimental Results. In: *IEEE Trans. Automat. Contr.* 44 (1999), Februar, S. 294–309
- [Tsai 1999] TSAI, L.-W.: *Robot Analysis, The Mechanics of Serial and Parallel Manipulators*. New York, USA : John Wiley and Sons, 1999
- [van der Schaft 2000] VAN DER SCHAFT, A. J.:  *$\mathcal{L}_2$ -Gain and Passivity Techniques in Nonlinear Control*. 2. Auflage. London, UK : Springer-Verlag, 2000 (Communications and Control Engineering)
- [VDMA 2002] VDMA: *Schlüssel zur Produktivität 2002. Robotik und Automation*. 2002
- [Vukobratovic und Tuneski 1994] VUKOBRATOVIC, M. ; TUNESKI, A.: Contact Control Concepts in Manipulation Robotics: An Overview. In: *IEEE Trans. Ind. Electron.* 41 (1994), Februar, Nr. 1, S. 12–24
- [Wang u. a. 2004] WANG, Z. P. ; GE, S. S. ; LEE, T. H.: Robust Motion/Force Control of Uncertain Holonomic/Nonholonomic Mechanical Systems. In: *IEEE/ASME Trans. Mechatron.* 9 (2004), März, Nr. 1, S. 118–123
- [Whitcomb u. a. 1993] WHITCOMB, L. L. ; RIZZI, A. A. ; KODITSCHKE, D. E.: Comparative Experiments with a New Adaptive Controller for Robot Arms. In: *IEEE Trans. Robot. Automat.* 9 (1993), Februar, Nr. 1, S. 59–70



- 
- [Yoshikawa 2000] YOSHIKAWA, T.: Force Control of Robot Manipulators. In: *Proc. IEEE Int. Conf. on Robotics & Automation (ICRA 2000)*. San Francisco, USA, April 2000, S. 220–226

